SUBHARMONIC ALIASING AND ITS EFFECTS ON THE AFTI/F-16 DIGITAL FLIGHT CONTROL SYSTEM

THESIS

David M. Thomas
Captain, USAF

AFIT/GE/ENG/87D-66

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/87D-66

SUBHARMONIC ALIASING AND ITS EFFECTS ON THE AFTI/F-16 DIGITAL FLIGHT CONTROL SYSTEM

THESIS

David M. Thomas
Captain, USAF

AFIT/GE/ENG/87D-66

Approved for public release; distribution unlimited

AFIT/GE/ENG/87D-66

SUBHARMONIC ALIASING AND ITS EFFECTS ON THE AFTI/F-16 DIGITAL FLIGHT CONTROL SYSTEM

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

David M. Thomas, B.S.
Captain, USAF

December 1987

Approved for public release; distribution unlimited

Preface

The purpose of this thesis was to analyze the causes of subharmonic aliasing, develop an analytic model of this effect,and determine its characteristics on the AFTI/F-16 Digital Flight Control System (DFCS).

Many hybrid (SIMSTAR) computer simulations were performed to verify that subharmonic aliasing existed and determine its effects on the asynchronous, triply redundant digital flight control system. After weeks of trying to model the effect in the time domain, I was steered toward the Fourier transform and the frequency domain by Lt Col Lewantowicz. Within days a model was formed.

In performing the simulations and modelling the phenomenon, I received a great deal of help from others. I would like to thank my thesis advisor, Lt Col Lewantowicz, for his guidance in helping me model subharmonic aliasing. I would also like to thank Bob Ewing for his help with SIMSTAR and for providing a work area used for the many hours of programming necessary to complete this research, and Don Smith (EAI) for his persistance in keeping the SIMSTAR "up". Finally, I would like to thank my family, Jeremy, Nathan, Bethany, and especially my wife Cindy, for their patience, love, and understanding throughout the entire eighteen months at AFIT.

David M. Thomas

Table of Contents

List of Figures

List of Tables

Abstract

The purpose of this research is threefold. First, determine the cause of subharmonic aliasing, described by the AFTI/F-16 engineers as "the creation of uncorrelated low frequencies whenever a subharmonic of the sample frequency is input into the system". Second, model the subharmonic aliasing effect, so that, by knowing only the input frequency and the system sample rate the output characteristics can be calculated. And third, demonstrate by simulation the effect of input and output filters on the subharmonic alias, and the effect of signals in the subharmonic range ($\omega_s/10 < \omega_o < \omega_s/2$) on the interchannel difference and the software rate limiter.

The model determined that subharmonic aliasing is the result of impostor frequencies (much like aliasing) being introduced into the output signal by the sampling process. Well defined subharmonic aliases occur due to: impostor frequencies and a phenomena known as apparent low frequency surge, which occurs when the input frequency is nearly an integer multiple (>1) of the impostor frequency.

Simulations demonstrate the effectiveness of input analog low pass filters for attenuating signals in this frequency range. Unfortunately, according to simulations,

interchannel differences may exceed Input or Output selector monitor threshold levels for frequencies in the subharmonic range. Simulations also show that it is the combination of well defined subharmonic aliases, the high frequencies (relative to the sample rate) of the input signal, and the nonlinear characteristics of the software rate limiter which form the low frequency output from the rate limiter.

In conclusion, since subharmonic aliasing occurs for frequencies which are high relative to the sample rate, careful selection of the sample rate (and computational frame rate) should reduce this undesireable effect. Additionally, since the phenomenon is actually the combination of two high frequencies, the use of analog lowpass filters at the input does attenuate the subharmonic aliasing effect.

SUBHARMONIC ALIASING AND ITS EFFECTS ON THE AFTI/F-16 DIGITAL FLIGHT CONTROL SYSTEM

I. Introduction

With the advent of smaller, faster computers came the pioneering of digital flight control systems (DFCS) to steer future aircraft (1:1). The Advanced Fighter Technology Integration (AFTI) F-16 is such an aircraft. Its asynchronous, triple-redundant, digital flight control system was designed to meet all flight control needs previously performed by a quad-redundant analog system. Because the AFTI/F-16 is completely fly-by-wire control (there are no mechanical backups to the flight control system), considerable effort was made to ensure reliability, safety, and performance. In case of hardware failure within the DFCS, an advanced strategy for redundancy management was developed to both detect and isolate DFCS failures.

Redundant elements (sensors,computers, etc) were designed into the system, and to monitor the inputs and outputs of these elements an advanced strategy for redundancy management was developed. In such a design, comparison monitoring and voting is used to detect, isolate, and remove failures. These redundant elements and the strategy for redundancy management will be described briefly in Chapter 2.

Two problems are inherent to the AFTI/F-16 DFCS: interchannel differences caused by time skew and subharmonic aliasing. The first problem is unique to the AFTI/F-16 since its DFCS is asynchronous. In effect, the asynchronous operation causes interchannel differences, because of time skew, between the three redundant channels. The interchannel difference may be interpreted by the fault detection and isolation algorithms as a failure. This problem is discussed extensively in Lt Gursel Serpen's thesis "Failure Detection and Isolation for an Asynchronous Digital Flight Control System" (2).

Problem

The second problem, the main topic of my thesis, is subharmonic aliasing. Subharmonic aliasing is a phenomenon described by the AFTI/F-16 engineers as the creation of uncorrelated low frequencies whenever a subharmonic of the sampling frequency (f_s) is input into the system (1:10). This study shows that it is actually a phenomena which effects all sample data systems that do not have ideal anti-aliasing filters. Subharmonic aliasing is most prominent for sinusiodal inputs, whose frequencies f_o (Hz) are less than $f_s/2$ (the Nyquist rate) but greater than $f_s/10$. Figures 1(a & b) show two examples of subharmonic aliases. The input is a sine wave, the output is the input sampled at 64 samples/second (the DFCS sample rate). A main concern of this thesis is that the AFTI/F-16 has a number of structral modes within a range of frequencies labeled the subharmonic alias range, $f_s/10$ Hz $< f_o < f_s/2$ Hz, which for $f_s = 64$ samples/second is 6.4 Hz $< f_o < 32$ Hz (see Table I).

Because of the asynchronous DFCS architecture of the AFTI/F-16, the high frequency component of the subharmonic aliased signal may cause large interchannel differences between redundant channels and may appear as a disagreement to the signal monitors. This disagreement may trigger the fault detection and isolation algorithm, which in turn may vote a sensor as failed when no failure actually exists. Abnormally high Input and Output selector monitor threshold levels are used to tolerate this in the AFTI/F-16. (This paragraph directly contradicts the AFTI/F-16 engineers assessment that it is the low frequency component that would cause the interchannel differences. The problem is explained in detail in Chapter 5.)

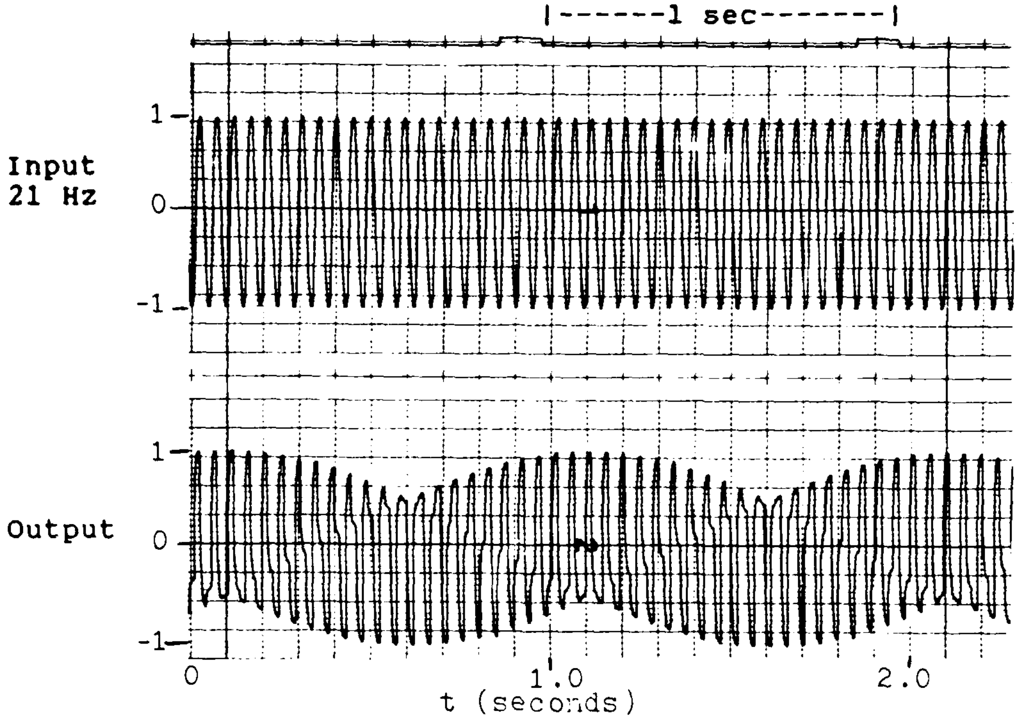

Figure 1a. Subharmonic Aliasing Effect (input 21 Hz, sampled at 64 samples/sec)

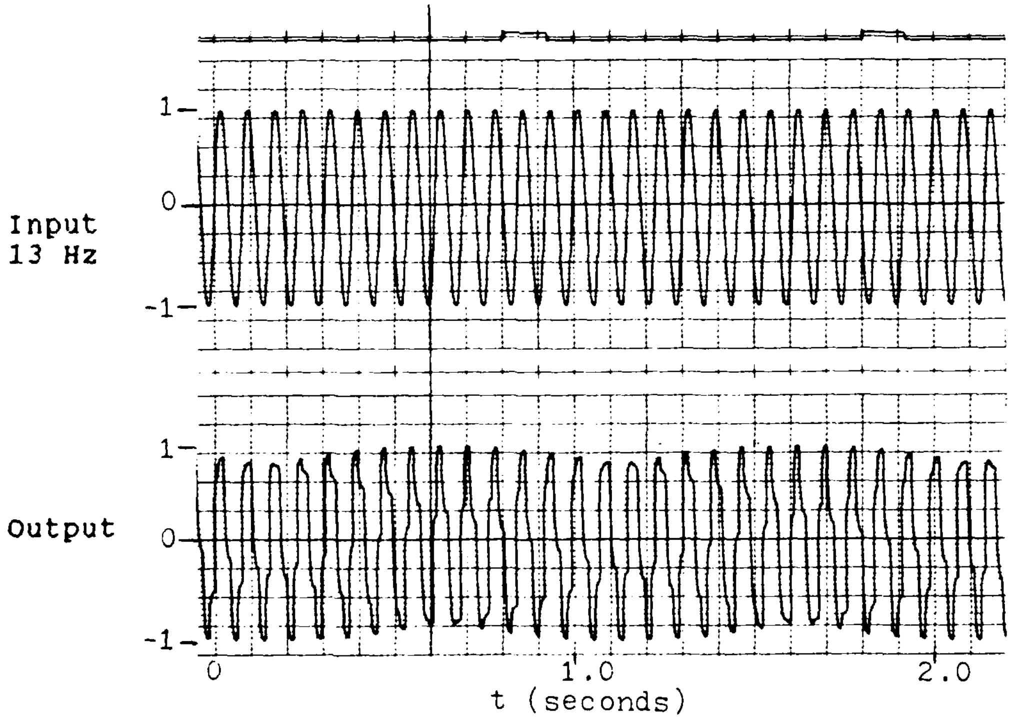

Figure 1b. Subharmonic Aliasing Effect (input 13 Hz, sampled at 64 samples/sec)

Table I. AFTI/F-16 Structural Modes

	Frequency (Hz)	Damping (g)
Symmetric	4.59	.023
	6.9	.018
	11.26	.040
	19.235	.042
	20.463	.056
	21.186	.040
	35.592	.056
	37.293	.020
Antisymmetric	5.798	.023
	8.343	.022
	12.753	.044
	15.887	.060
	19.177	.057
	22.008	.046

Assumptions

The major assumption for this thesis is that, although the control laws and aircraft dynamic model are not simulated, the sample rates and inputs are simulated and this, adequately demonstrates subharmonic aliasing and its effects on the DFCS.

Scope

This thesis describes in detail, mathematically and by simulation, subharmonic aliasing. It shows how frequencies in the "subharmonic aliasing range" effect the interchannel differences between the Digital Flight Control Computers because of time skew and how the actual threshold levels for the Input S/M and the Output S/M can be violated. Finally, effects of the rate limiter are demonstrated and two configurations of the rate limiter placement within the DFCS are considered.

In all hybrid computer (SIMSTAR) simulations, inputs are generated by an internal harmonic oscillator (sine wave generator) rather than external inputs to limit noise effects and for ease of parameter control.

Approach

The following approach is used to investigate the "Effects of Subharmonic Aliasing on the AFTI F-16 Digital Flight Control System".

1. A simple sampling system is simulated on the hybrid computer (SIMSTAR) to recreate the subharmonic aliasing effect as described by the AFTI/F-16 engineers.

2. A mathematical model is developed and used to predict the magnitude and frequency of the subharmonic aliased signal. A digital computer simulation package is used to verify the model.

3. The maximum interchannel difference is mathmatically computed and graphed for the frequencies in the subharmonic range. Hybrid and digital computer simulations are used to verify the computations.

4. A software rate limiter is simulated using a FORTRAN subroutine in a hybrid computer to determine its effect on subhamonic aliased signals.

5. Means to reduce subhamonic aliasing are examined.

6. Follow-on work is suggested.

The Resources required to accomplish this thesis are listed in Appendix C.

II. BACKGROUND

The AFTI/F-16 Digital Flight Control System

The AFTI/F-16 Digital Flight Control System (DFCS) is a full authority, triplex, digital fly-by-wire flight control system that includes provisions for stringent reliability and fail-safe operational requirements (3:4-1). The triplex configuration consists of three identical flight control computers (FLCC) which operate asynchronously with respect to each other. They are designed to operate mutually independent of each other for increased system reliability. Each FLCC treats the other two as remote sensors rather than as an external computational source, and each FLCC forms an independent view as to the state of the system. Therefore, it is possible for one computer to differ from the others with respect to what is failed in the system or in what mode the system is operating (3:1-3).

The overall DFCS is illustrated in Figure 2. The three identical FLCC's provide computational power for control law computation and redundancy management. A fourth unit (not shown), the Actuator Interface Unit (AIU), houses dual-redundant analog functions associated with the leading-edge flap actuator drives (3:4-23).

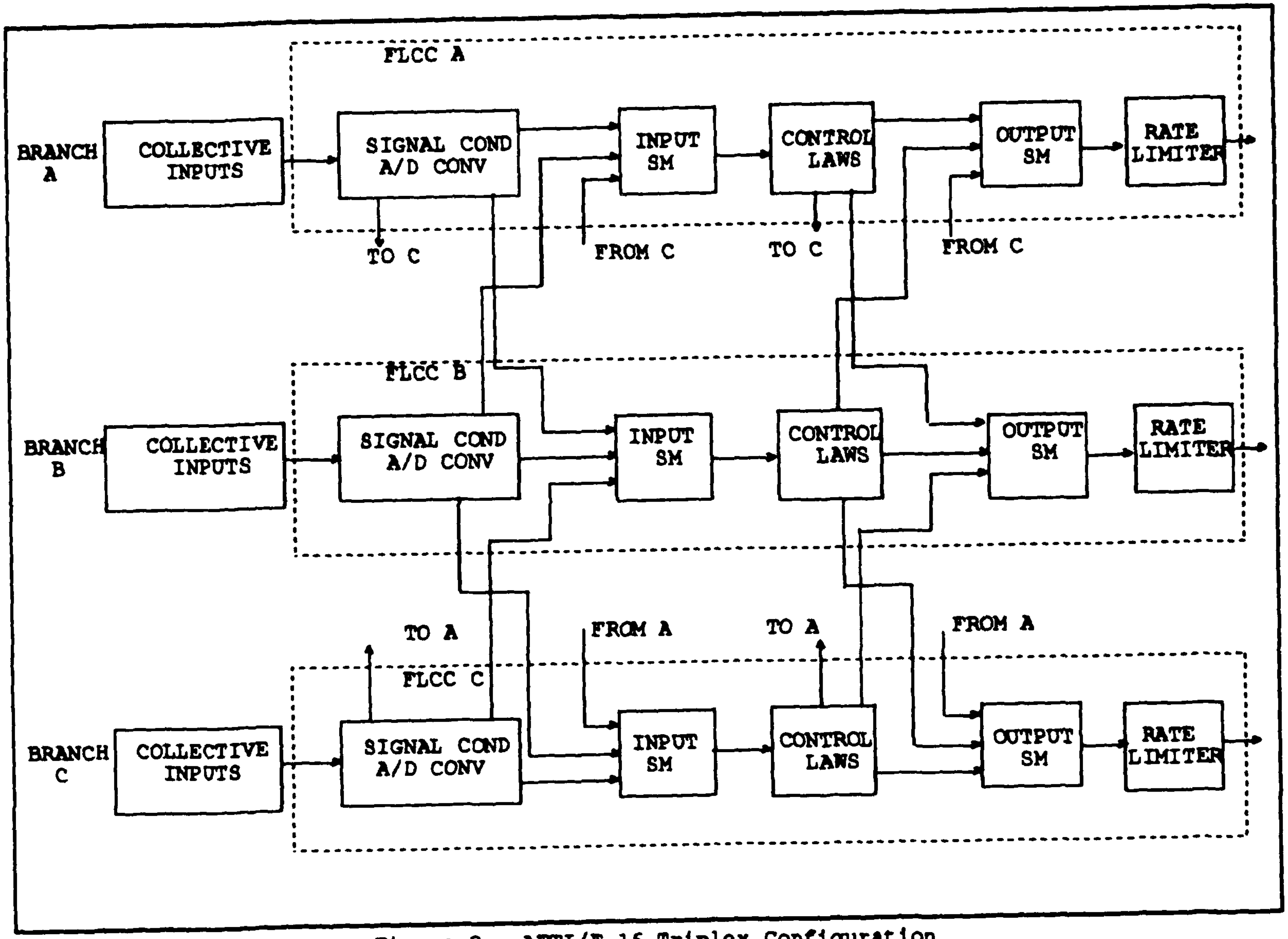

Figure 2. AFTI/F-16 Triplex Configuration

The Computer Architecture

The highlights of the computer architecture are presented here, while detailed information is contained in Reference 3. Each FLCC contains the following eight major functional sections:

1. Analog/discrete input signal conditioning and multiplexing
2. Analog/discrete output signal conditioning and demultiplexing
3. Digital computation and data conversions
4. Intercomputer data link interface
5. Multiplex bus (MIL - STD - 1553) interface
6. Dedicated failure logic
7. ISA drives, associated switching, and warning light drives
8. Analog independent backup unit (IBU)

The digital portion of the computer architecture is composed of the following five basic sections:

1. Central Processing Unit (CPU)
2. Memory (CPU Random Access Memory)
3. Input/output controller (IOC)
4. Intercomputer data link receiver controller
5. The 1553 Multiplex bus interface controller

In this architecture, the processor communicates only with the memory and performs no input/output (I/O) operations except for those critical discretes that are mapped into memory (3:4-29).

Input/Output Controller

The I/O operations are controlled by the Input/Output Controller (IOC) which minimizes the CPU real time processing requirements. The IOC supervises and controls the operation of the Input/Output Multiplexer/Demultiplexer system, the analog to digital (A/D) and digital to analog (D/A) converters, the Intercomputer Data Link Transmitter and the Input/Output Discrete Registers. The IOC is microprogrammable to allow a random sequence of I/O operations (up to 1000 unique operations), and in the autonomous mode its operation is completely transparent to the processor. For normal (non-failure) conditions, the processor retains supervisory authority over the operation in that it can vector the controller to specific prepro-grammed I/O sequences, and it receives an indication from the IOC that the requested sequence has been completed. This provides reasonable control over the latency of data to be used for the control law or redundancy management functions. The individual operations of the command sequence, however, remain transparent to the processor. The IOC selects the I/O device, initializes conversions and transfers data to/

from memory according to a preprogrammed file, hence the processor needs only to access the memory for I/O operations. For detected failures of the channel (e.g., the processor), the dedicated failure logic overrides the processor control and vectors the I/O controller to a special file which maintains the unit in a free-running mode until the failure condition is removed. This free running autonomous mode, allows the cross-channel communication of sensor data (and fault status) to be continued to the other two channels of the triplex system even after a processor failure (4:1429).

Monitoring, Voting, and Failure Management

Most of the monitoring, voting, and redundancy managegement functions within the flight control computer are implemented in software. Voter/monitor planes are provided in each channel for redundant sensor inputs by use of digitally cross-strapped data from the other two channels. Cross-channel comparison monitoring of sensor and computed data is the primary method of fault detection with in-line self-testing used only to isolate a failed channel following a second similar failure detected by the cross-channel monitoring. This method achieves a probability of loss-of-control (PLOC) of 10^{-7} by the use of only three channels of computation (5:1222).

Provisions are included to assure that a failed branch is prohibited from contributing to the control of the aircraft. Since a malfunctioning computer cannot be expected to recognize its own errors, special dedicated hardware isincluded to permit the unfailed processors to disconnect the malfunctioning computers outputs. For a second like-failure control, each remaining processor can alarm the failure and require a response. The AFTI/F-16 uses this approach since internal self-testing is relied on for action in resolving second failures rather than empower a single branch to disconnect another branch. If the second like-failure cannot be resolved by self-testing, automatic reversion to an analog backup control system is provided (3:4-41).

Input Selector Monitor Plane (Input S/M). The function of the input selector monitor is to detect failures of analog inputs from the three redundant FLCC's and determine a good analog input for use by the control law function. Failure detection is accomplished by use of cross-channel monitoring techniques (6:3-9).

Note:

Self - refers to the FLCC performing the calculation.
Left and Right - refer to the "other two" FLCC's.

Analog inputs are monitored in the following manner. The current value of a sensor input is read from the self IOC random access memory (IOC RAM) (Figure 3) and the latest

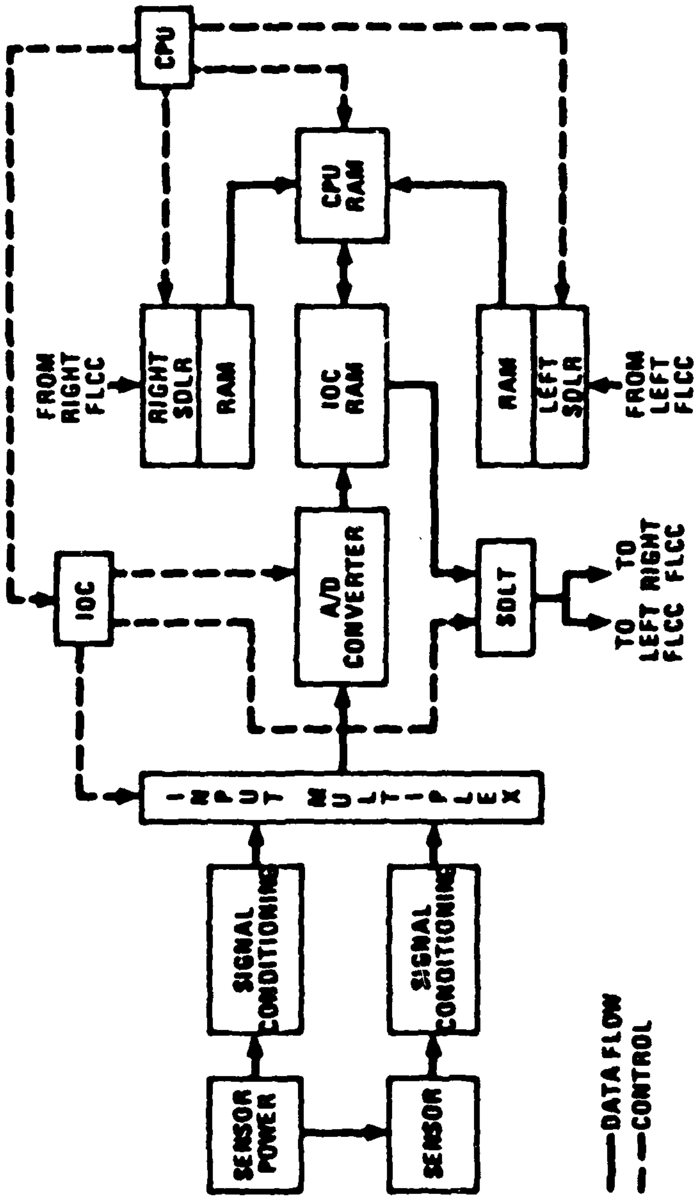

Figure 3. Input Signal Flow/Hardware Relationsips (5:1223)

value from the left and right serial data link receiver (SDLR) RAM areas respectively. If the three values differ by no more than the allowable interbranch trip level for this parameter (15% of full scale for most parameters), then the average of these three values is calculated and stored in the CPU RAM area for later utilization by the control law functions. If one of the three input values differs from the other two an amount greater than the trip level, the out of tolerance signal is rejected by the monitor and the selector algorithm calculates the average of the remaining two good values. This type of algorithm is known as a good channel averager (3:1223).

If this out of tolerance condition persists for more than seven major computer frames (each frame is 15.625 msec) the failure is considered a hard failure and a call is made to the Failure Management function to record the failure and to perform further failure analysis.

Output Selector Monitor. The function of the Output Selector Monitor (Output S/M) is to monitor the Total Computed Output (TCO) (Figure 4) or surface commands to detect computational failures and to ensure that the three coil current drives to each ISA servo valve are always within the ISA hydraulic trip level (10% to 30% of full scale coil current) so long as the command path is failure free. The design of the output monitor is similar to the Input S/M design and includes three persistence counters, one totalizer counter, separate first fail and second fail

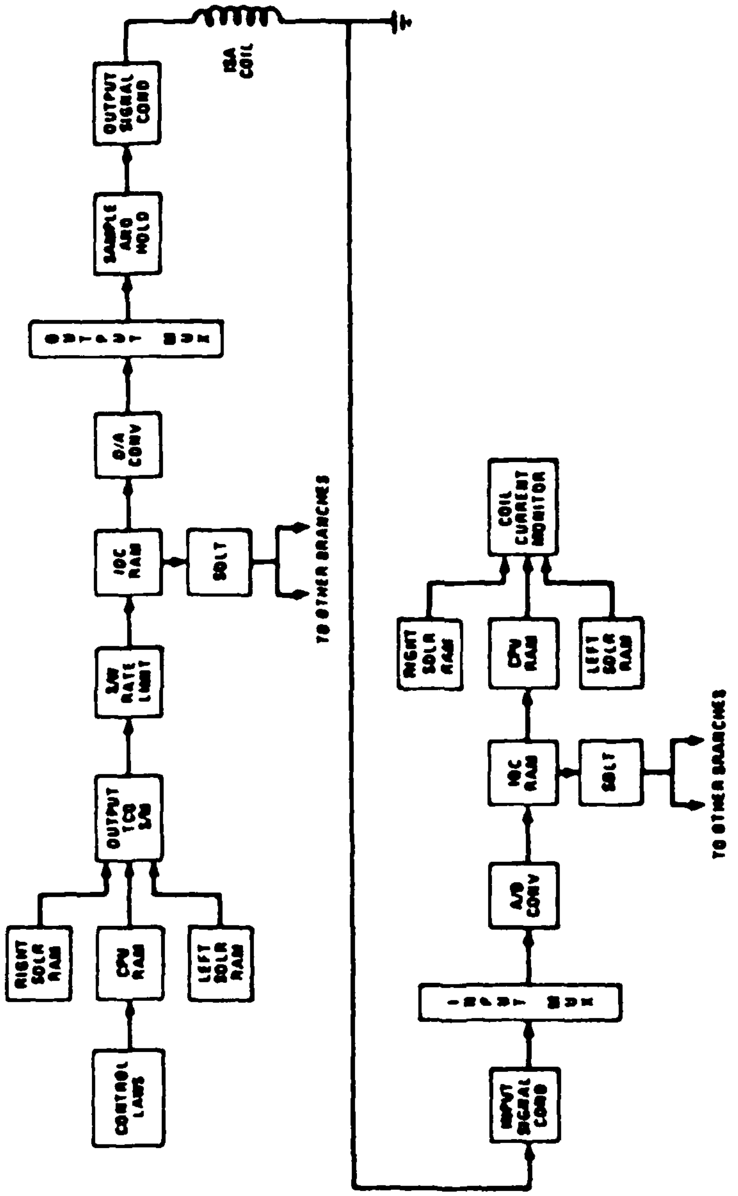

Figure 4. Output Signal Flow/Hardware Relationships (5:1225)

maximum persistence values, and a fixed allowable cross-channel difference. In order to minimize latency through the selector, it is necessary that the selector pick the output of the computer that is actually driving the surface. When the system is operating from the primary hydraulic system (servo valve 3), minimum latency is achieved by selecting FLCC C to drive the system (6:5-39). (For configuration management purposes the FLCC's are identified as FLCC A, FLCC B, and FLCC C).

Failure Management. The Failure Management function provides centralized management of the DFCS failures. This function checks the status of the DFCS by referring to the Device Status Table (DST), it records/updates the device failure on the DST to remove the failed element from the redundancy scheme, and it maintains a count of low level device failures to determine if a high level device is actually failed (6:11-1,11-19).

Data Acquisition

Several key factors are important in developing the overall data acquisition scheme for the AFTI/F-16 DFCS. Only one of these factors is presented here. The major factor that determines the rate at which data must be acquired is the rate of change of the input signal with respect to time. In general, if the rate of change of the input is high relative to the sampling rate, a large interchannel difference is generated in otherwise identical signals when the data is exchanged between computers (6:2-1).

For example, the largest pilot stick pitch input (s) expected is 0 to 9 (full scale) in 0.1 seconds, in other words an input rate of 90 g/second ($\dot{s}$). The maximum sample period (T) allowed to keep the interchannel difference (e) less than 5% can be calculated by the simple equation

$$\begin{aligned} T &= e_{max}/\dot{s} \qquad (1) \\ &= 0.45\ g/(90\ g/sec) \\ &= 0.005\ second \end{aligned}$$

where

T = period

e_{max} = maximum interchannel difference

= 9 g * 0.05

= 0.45 g

$\dot{s}$ = maximum signal rate of change (time derivative)

= 9 g / 0.1sec

= 90 g/sec

Since the DFCS has a computational frame period of 1/64 seconds or 15.625 msec (6:2-8), for ease of implementation, a multiple of this period is required for input data acquisition. The input period selected is 3.90625 msec

(15.625/4) or 256 samples per second.

As shown in Figure 5 (6:2-3) this rate is adequate even when combined with anti-aliasing filters or prefilters with cutoff frequencies at ω = 100, 50 and 10 rad/sec. However, as break frequencies decrease, excessive phase lags are introduced resulting in reduced gain and phase margins and poor handling characteristics (6:2-2).

Since interchannel differences are not presently reduced by voter algorithms (as are sensor bias and calibration gain errors), minimizing these errors with an adequate sampling rate is critical for the AFTI/F-16 (6:2-2). (See Ref 2, Lt Gursel Serpen's thesis, for a method to reduce the interchannel differece by estimating the time skew.)

Rate Limiters

Software rate limiters are mechanized in the surface command mixer just after the Output S/M plane (Figure 4) for each control surface except for the leading edge flap. The rate limits are set to limit the signal change per iteration to each surface to no more than 5% of full scale deflection. The purpose of this rate limiting is twofold. First, it acts to minimize interchannel differences caused by time skew, and second, it acts to prevent erroneous ISA output electronic trips due to large frame-to-frame command deviations (6:13-23).

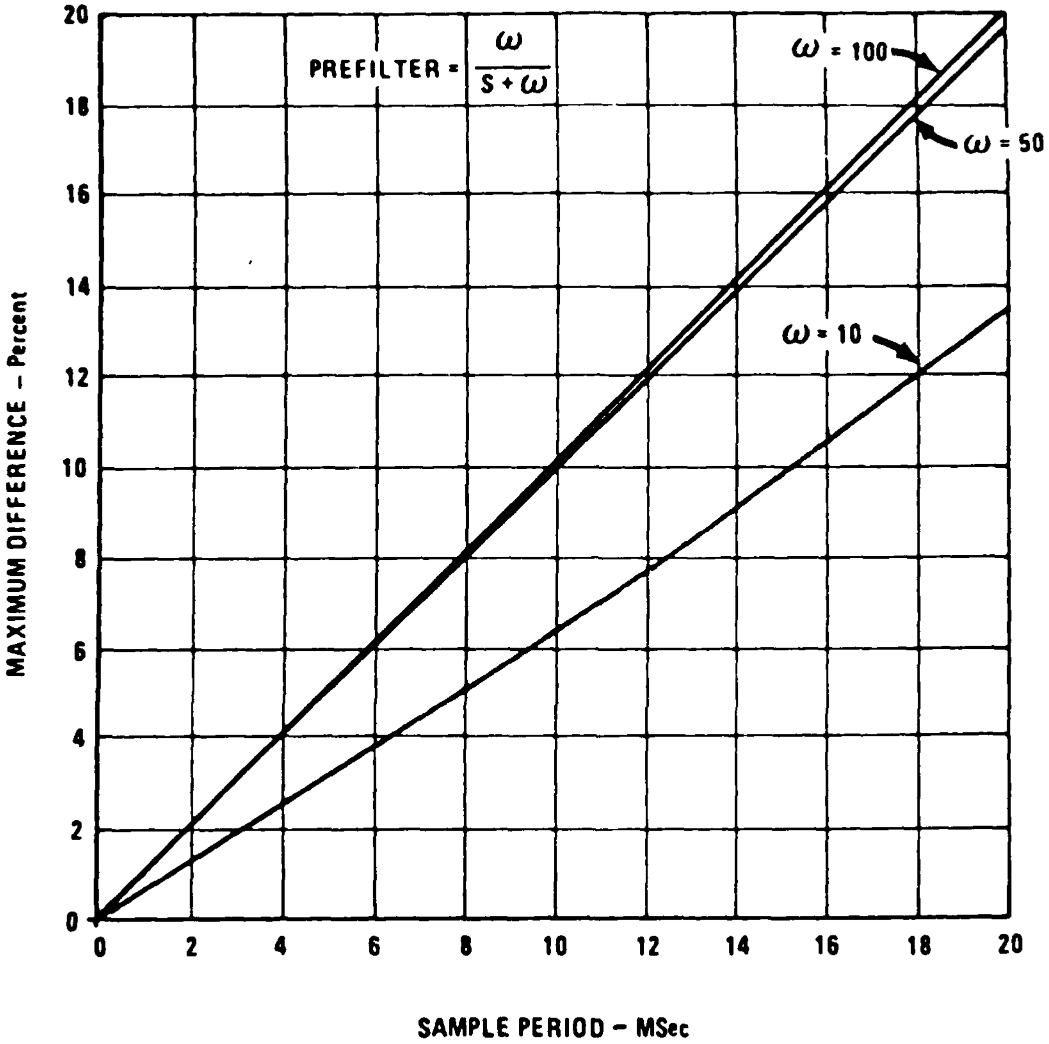

Figure 5. Interchannel Difference for 90 g/second Ramp (6:2-3)

Aliasing

Aliasing is a phenomena well described in numerous texts (8,9,10) on digital control and communications and can be found under headings of either "aliasing" or "folding". It occurs when Shannons Sampling Theorem is violated. Shannons Sampling Theorem simply states:

> Let x(t) be a bandlimited signal with $X(\omega) = 0$ for $|\omega| \leq \omega_m$. Then x(t) is uniquily determined by its samples x(nT), n=0, ±1, ±2,... if
>
> $$\omega_s > 2\omega_m \qquad (2)$$
>
> where
>
> $\omega_s = 2\pi/T$ sample radian frequency
> T = sample period
> ω_m = maximum radian frequency
>
> Given these samples, x(t) can be reconstructed by generating a periodic impulse train in which successive impulses have amplitudes that are successive sample values. This impulse train is then processed through an ideal low-pass filter with gain T and cutoff frequency greater than ω_m and less than $(\omega_s - \omega_m)$. (7:519).

This theorem is violated when

$$\omega_m > \omega_s/2$$

where $\omega_s/2$ is refered to as the Nyquist rate, the rate which must not be exceeded by ω_m to prevent aliasing.

For example let

$$x(t) = \cos(\omega_o t) \qquad (3)$$

where

$\omega_o = \omega_m$

with Fourier transform $X(\omega)$ as indicated in Figure 6 (9:528). In this figure, the impulse is distinquished at ω_o from that at $-\omega_o$ for convenience, as the discussion proceeds. Consider $X_p(\omega)$, the spectrum of the sampled signal and focus in particular on the effect of a change in the input frequency ω_o with the sampling frequency ω_s fixed. Figure 6 (b)-(e) illustrates several values for ω_o. Also, indicated by the dashed line, is the ideal passband of a low-pass filter with $\omega_c = \omega_s/2$. Note that no aliasing occurs in (b) or (c), since $\omega_o < \omega_s/2$, whereas aliasing does occur in (d) and (e). For each of the four cases, the ideal low-pass filter output of $x_r(t)$ is given by:

b) $\omega_o = \omega_s/6;$ $x_r(t) = \cos(\omega_o t) = x(t)$

c) $\omega_o = 2\omega_s/6;$ $x_r(t) = \cos(\omega_o t) = x(t)$

d) $\omega_o = 4\omega_s/6;$ $x_r(t) = \cos(\omega_s - \omega_o)t \neq x(t)$

e) $\omega_o = 5\omega_s/6;$ $x_r(t) = \cos(\omega_s - \omega_o)t \neq x(t)$

When aliasing does occur, the original input frequency ω_o takes on the identity or "alias" of a lower frequency ω_A (7:527-529)

$$\omega_A = \omega_s - \omega_o \tag{4}$$

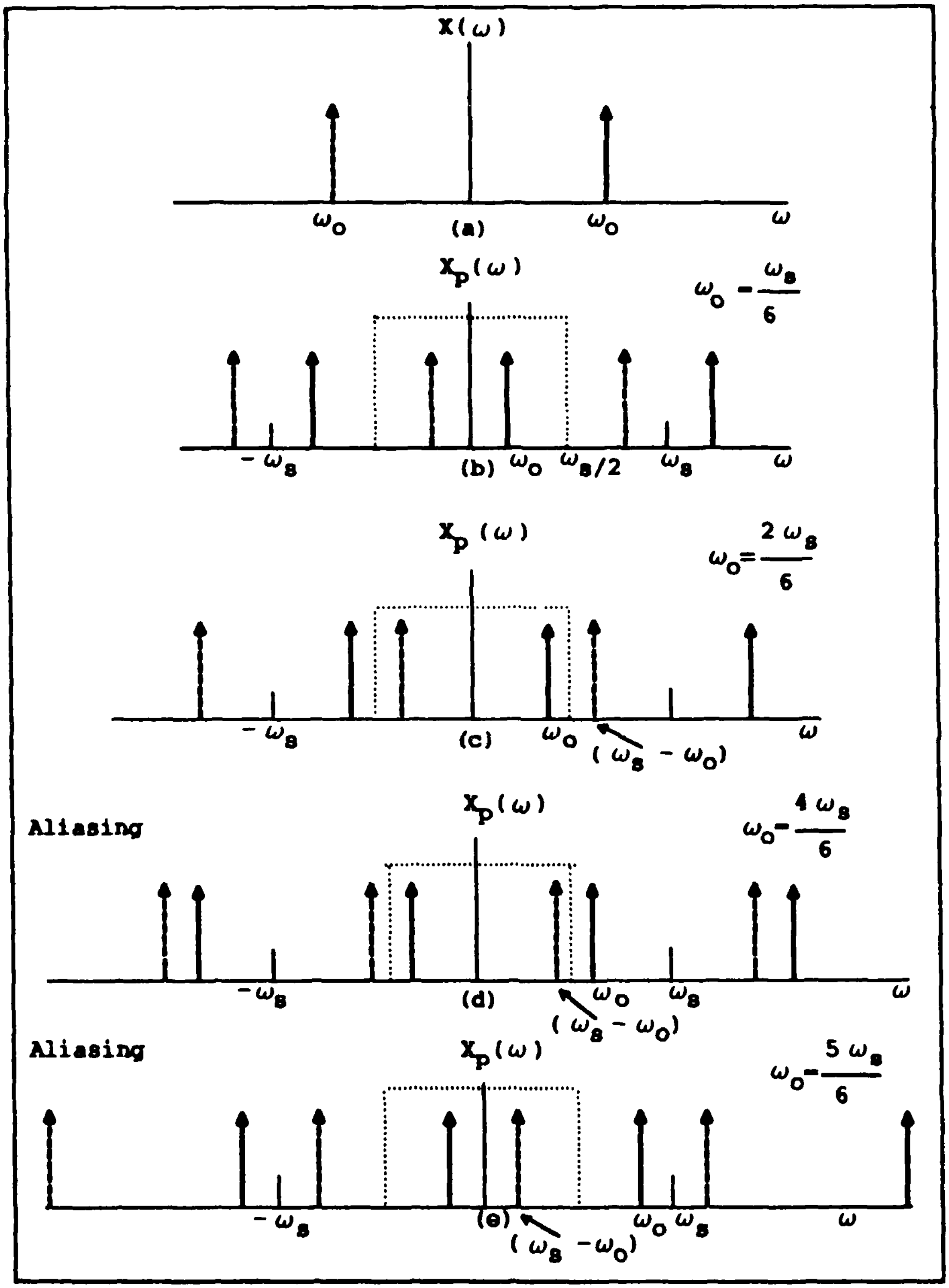

Figure 6. Effect in the Frequency Domain of Adequate Sampling (b) & (c), and Undersampling (d) &(e)

III. Simulation Model

Figure 7 is a model of the pertinent hardware/software relationships of the AFTI/F-16 digital flight control system which are used to study the effects of frequencies in the subharmonic range. For this study the subharmonic range is defined as the frequencies ω_0 where $\omega_s/10 < \omega_0 < \omega_s/2$. Inputs are generated with a sign wave generator that is internal to the hybrid (SIMSTAR) computer. Values for the anti-aliasing filters and prefilters, if used in the simulation, are from Table II which is a list of anti-aliasing and pre-filters used by the AFTI/F-16 (14). An input sample rate of 256 samples/second is used to simulate the sensor sample rate of the aircraft. The second sample rate of 64 samples/ second simulates the computational frame rate of the FLCC described in Chapter II.

Chapters 4, 5, and 6 implement different parts of this model to show

1. What subharmonic aliasing is.
2. How frequencies in the subharmonic range affect the magnitude of the interchannel difference.
3. How and why the rate limiter 'follows' the low frequency component of the subharmonic aliased signal and where the rate limiter should be (before or after the Output Selector Monitor (S/M).

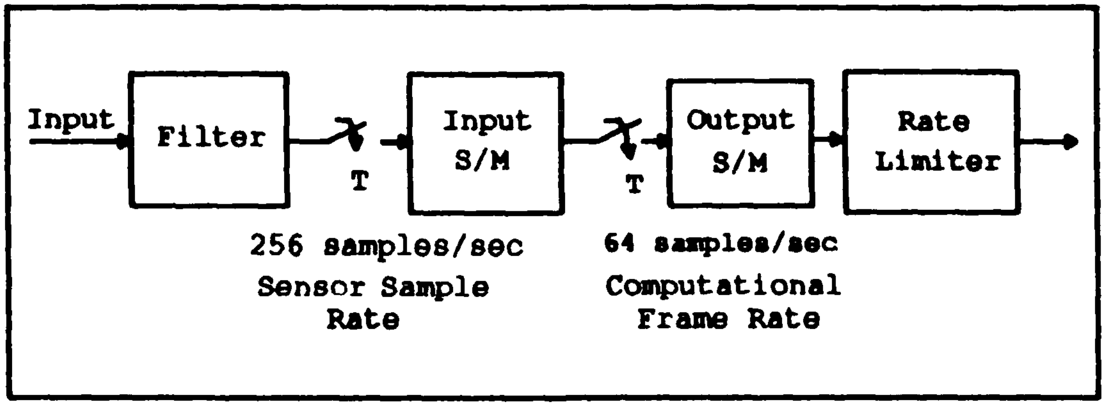

Figure 7. Hardware/Software Model

Table II. Selected Analog Anti-Aliasing and Pre-Filters

Mode	Filter(s)
Stick Input	$\frac{295}{(s+295)} * \frac{112}{(s+112)}$
Normal Accel (Anti-Alias)	$\frac{64.2}{(s+64.2)}$
Pitch Rate	$\frac{325}{(s+325)} * \frac{98.6}{(s+98.6)}$
Angle of Attack Yaw Rate Pitch Rate	$\frac{321}{(s+321)} * \frac{85.7}{(s+85.7)}$
Lateral Accel (Anti-Alias)	$\frac{96.2}{(s+96.2)}$
Rudder Pedal Throttle Twist Grip	$\frac{295}{(s+295)} * \frac{112}{(s+112)}$
Impact Pressure (Anti-Alias)	$\frac{96.2}{(s+96.2)}$

IV. Subharmonic Aliasing

Subharmonic aliasing is a phenomena dual to aliasing and associated with non-ideal sample data systems, that is, systems without ideal anti-aliasing filters with cutoff frequencies at $\omega_s/2$, which use a hold device (usually a zero-order hold) at its output. The purpose of this chapter is to describe and model subharmonic aliasing. It also provides a number of examples of subharmonic aliasing and the model, provides a means for predicting the magnitude and frequency of the subharmonic aliased signal, and shows the utility of the model for prediciting the output characteristics of a system.

Description of Subharmonic Aliasing

Given the simple circuit shown in Figure 8, let $x(t) = M * \cos(\omega_o * t)$. The Fourier transform $X(\omega)$ of the input is shown in Figure 9(a) and the frequency spectrum of the sampled input $X\ (\omega)$ is shown in Figure 9(b). This shows that there are components of the input signal not only at ω_o, but also at $n\omega_s \pm \omega_o$, where $n = 0, \pm 1, \pm 2, \ldots$.

Next, given the zero-order hold (ZOH)

$$G_{ho} = (1-\exp(-Ts))/s \qquad (5)$$

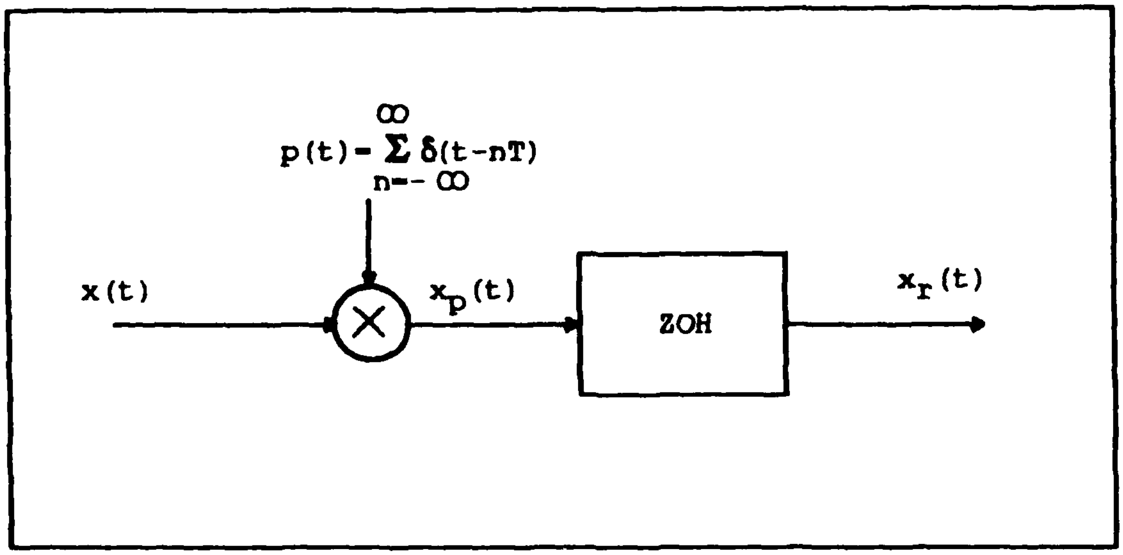

Figure 8. Sample-Data System

the frequency response can be found by replacing s by $j\omega$

$$G_{ho} = (1-\exp(-Tj\omega))/j\omega \tag{6}$$

which, after much mathmatical manipulation (6) can be written as

$$G_{ho} = \frac{T * \sin(\pi\omega/\omega_s) * \exp-j\pi(\omega/\omega_s)}{\pi\omega/\omega_s} \tag{7}$$

The gain characteristics, shown in Figure 9(c), imply that the ZOH behaves as a low pass filter. However, when compared to the gain characteristics of the ideal filter (Fig. 6, Chapter Two), the amplitude response of the ZOH is

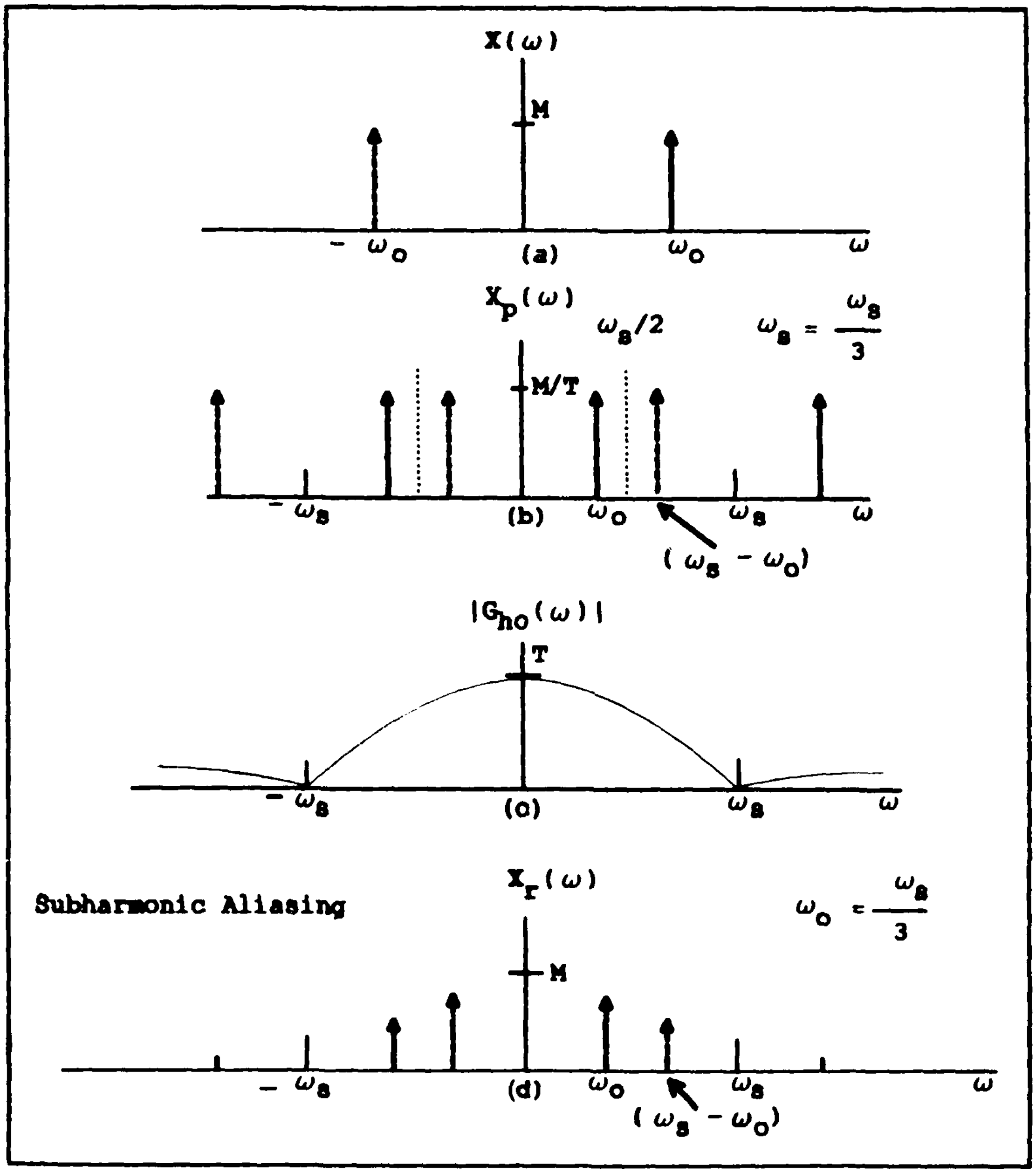

Figure 9. Subharmonic Aliasing in the Frequency Domain

zero at $\omega = \omega_s$, instead of cutting off sharply at $\omega_s/2$. This provides the impetus for the subharmonic aliasing phenomena.

The magnitude and phase characteristics of the ZOH are given by (11:134-135)

$$\phi(\omega) = \frac{-\pi\omega}{\omega_s} \qquad (8)$$

$$|G_{ho}(j\omega)| = \frac{T\ |\sin(\pi\omega/\omega_s)|}{\pi\omega/\omega_s} \qquad (9)$$

Note that the phase of the ZOH is linear with ω_s, and the gain is described by the sinc function.

When the composite spectrum of Figure 9(a) is filtered by the ZOH of Figure 9(c), impostor frequencies, shown in Figure 9(d), appear in the output signal (12:85), and create, as the AFTI/F-16 engineeres call them, subharmonic aliases.

Examples of Subharmonic Aliasing

The previous section describes subharmonic alaising. It is not, as was feared by the AFTI engineers, a problem to the AFTI/F-16 DFCS. This section simply provides a number of examples of subharmonic aliasing, generated by a hybrid computer (see Appendix B, SIMSTAR PROGRAMS). The simple A/D to D/A circuit in Figure 10 is used to generate

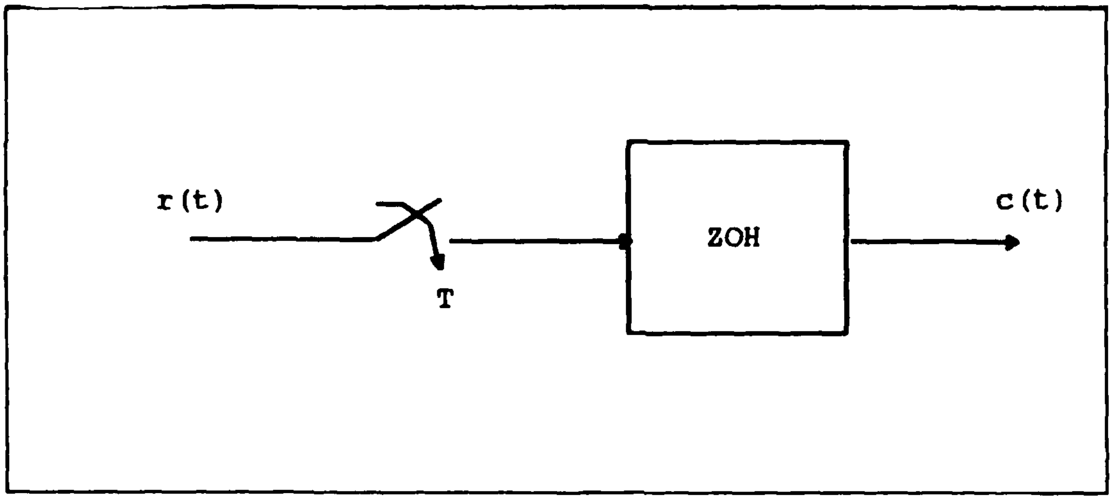

Figure 10. Subharmonic Aliasing Simulation

input signals which, when sampled, create impostors frequencies.

For an input

$$r(t) = \sin(\omega_o * t) \qquad (10)$$

The sampled output

$$C^*(s) = R^*(s)^*ZOH$$

$$C(z) = R(z)^*ZOH$$

$$c(kT) = \sin(\omega_o * k*T) \qquad (11)$$

where

ω_o < $\omega_s/2$ (the Nyquist rate)

ω_s = the sample rate (radians/second)

T = $2\pi/\omega_s$ in seconds

k = 0,1,2,...

This simulation uses a sample rate of 64 samples/second (the AFTI/F-16 control law computational frame rate). Figures 11-16 show how the impostor frequencies ($\omega_S - \omega_O$) effect the sampled input. For more examples on how the impostor frequencies affect the input see Appendix A.

The Model

Subharmonic aliasing is modeled using the information from figure 9(d), which shows how an impostor frequency is introduced into the sampled signal due to non-ideal anti-aliasing filters or the ZOH. As ω_O becomes small with respect to $\omega_S/2$ (as demonstrated in Figures A1-A7, in Appendix A) the output signal approaches a function of only ω_O since the impostor frequency $\omega_S - \omega_O$, which creates the subharmonic alias, is attenuated to nearly zero magnitude by the filter or ZOH. Given this information, a model of the sampled signal

$$c(kT) = M*\cos(\omega_O*k*T) \qquad (12)$$

where

$\omega_O < \omega_S/2$

which adequately describes the frequency and magnitude of the subharmonic alias is given by

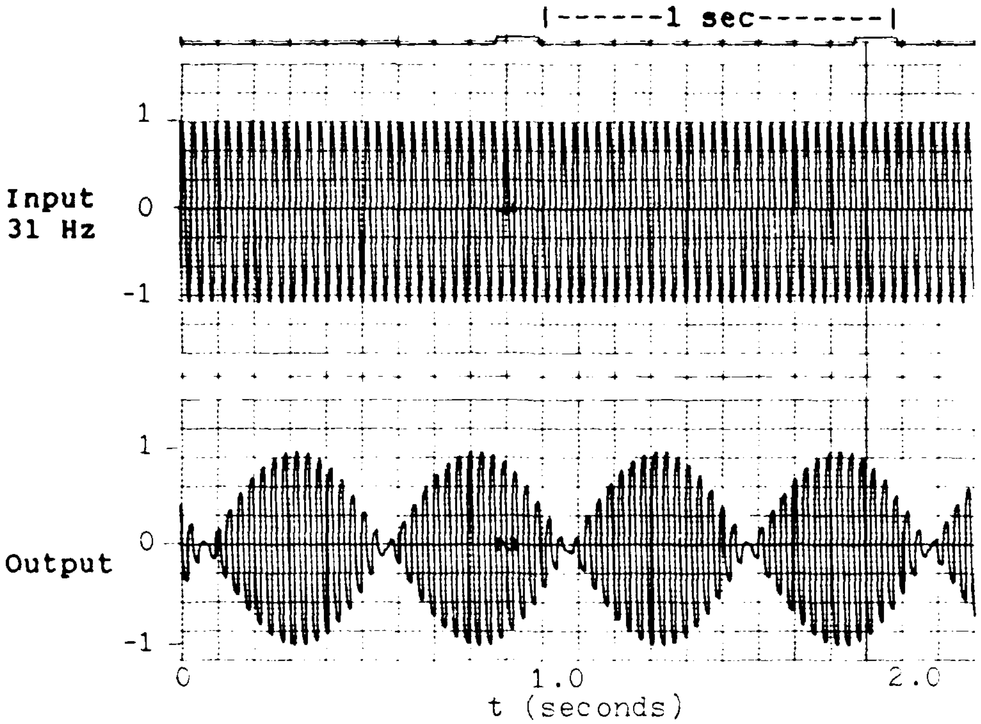

Figure 11. Subharmonic Aliasing Effect (input 31 Hz, sampled at 64 samples/sec)

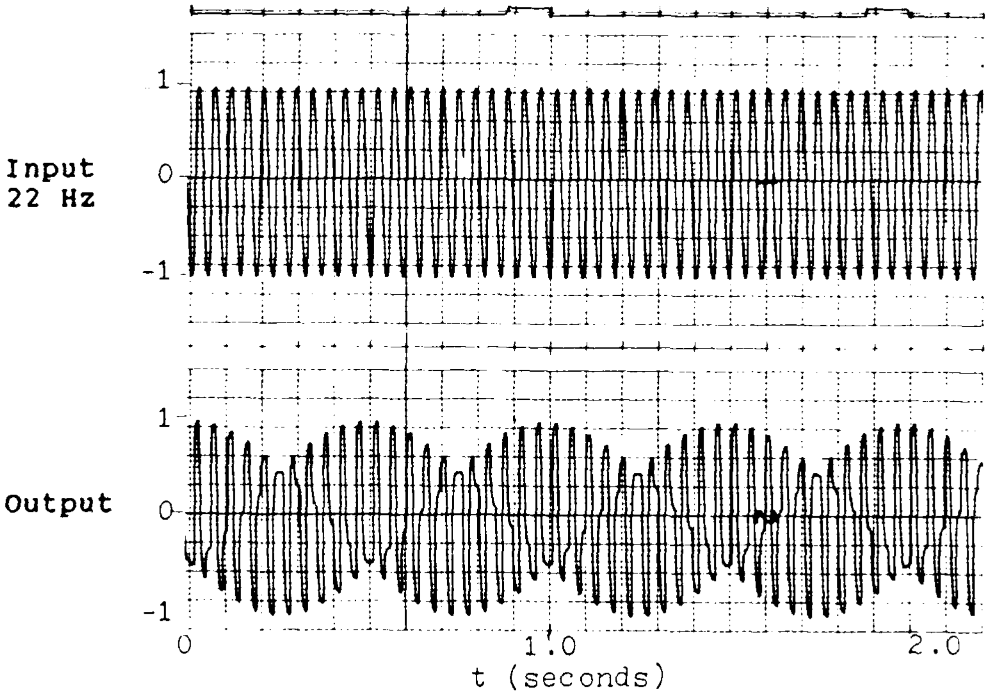

Figure 12. Subharmonic Aliasing Effect (input 22 Hz, sampled at 64 samples/sec)

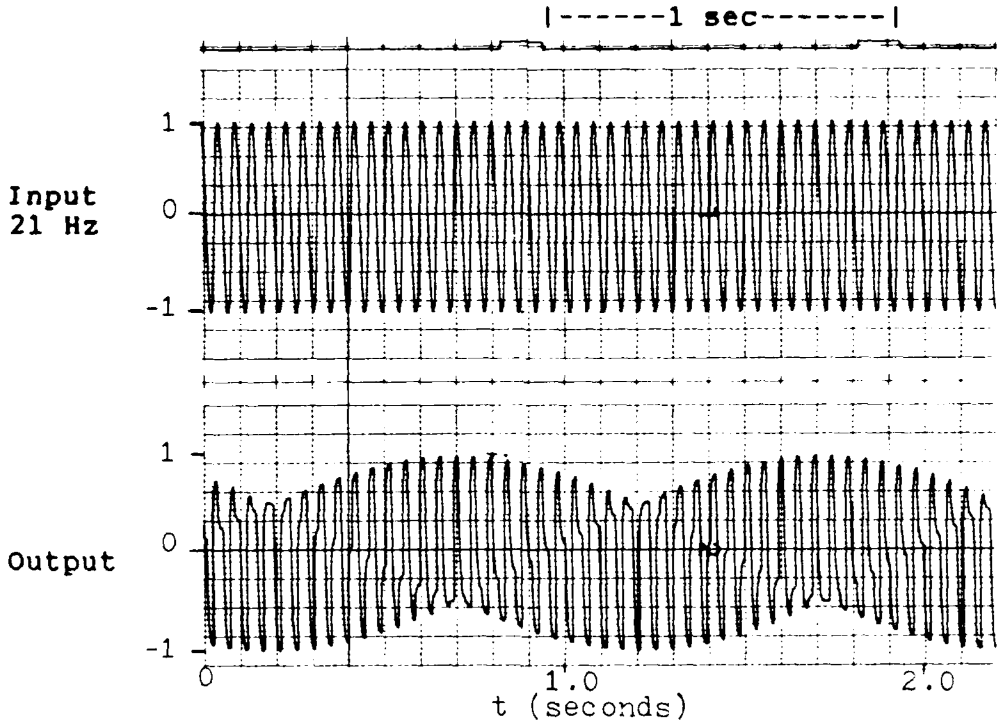

Figure 13. Subharmonic Aliasing Effect (input 21 Hz, sampled at 64 samples/sec)

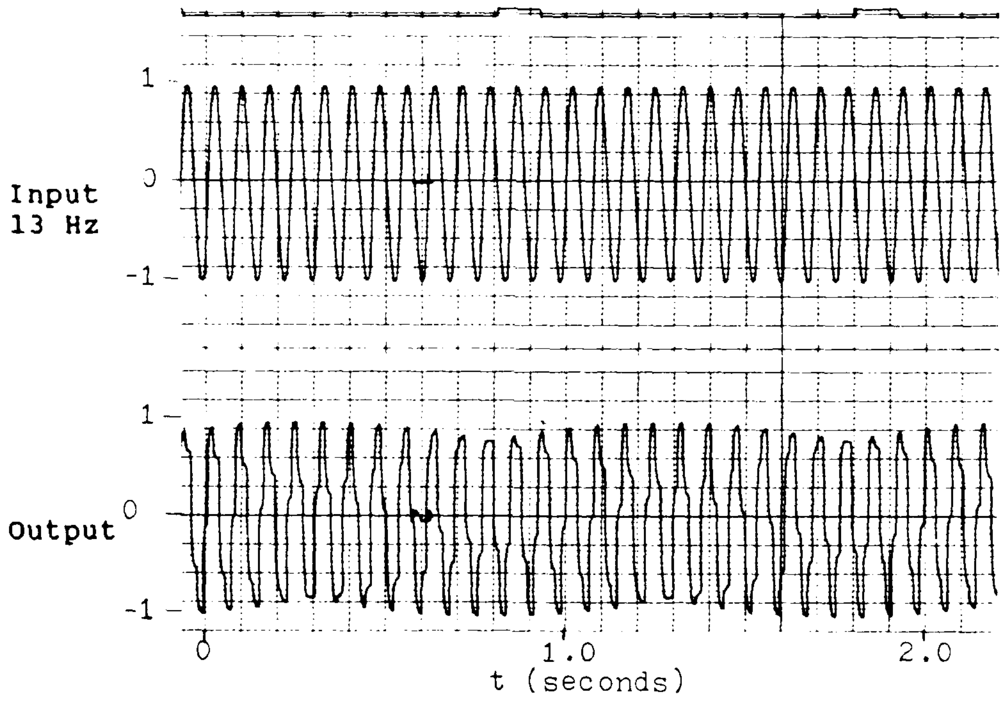

Figure 14. Subharmonic Aliasing Effect (input 13 Hz, sampled at 64 samples/sec)

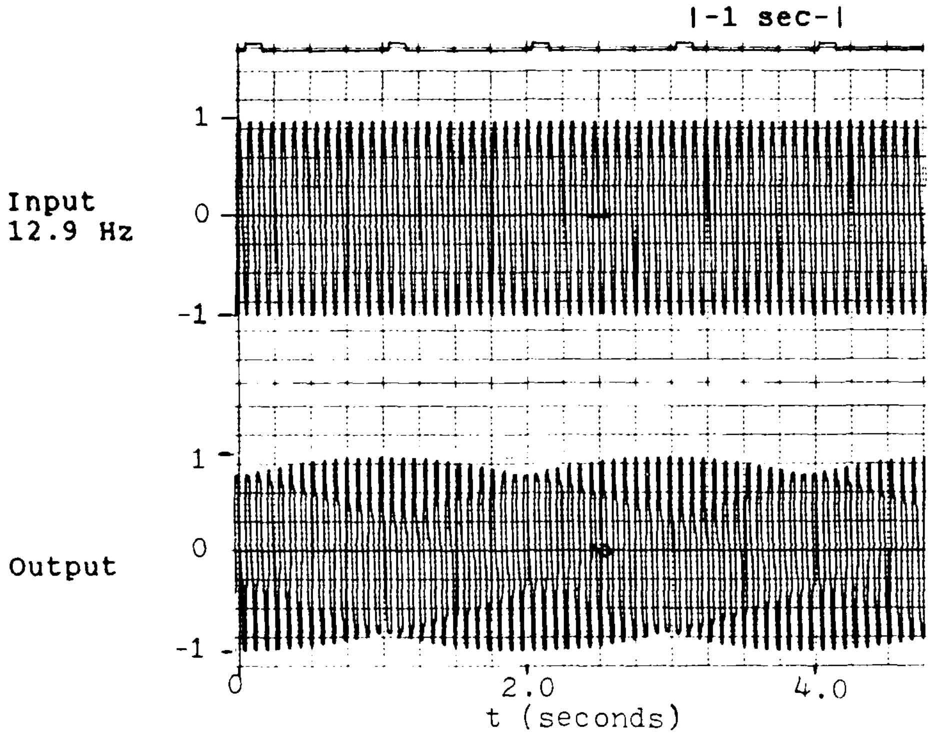

Figure 15. Subharmonic Aliasing Effect (input 12.9 Hz, sampled at 64 samples/sec)

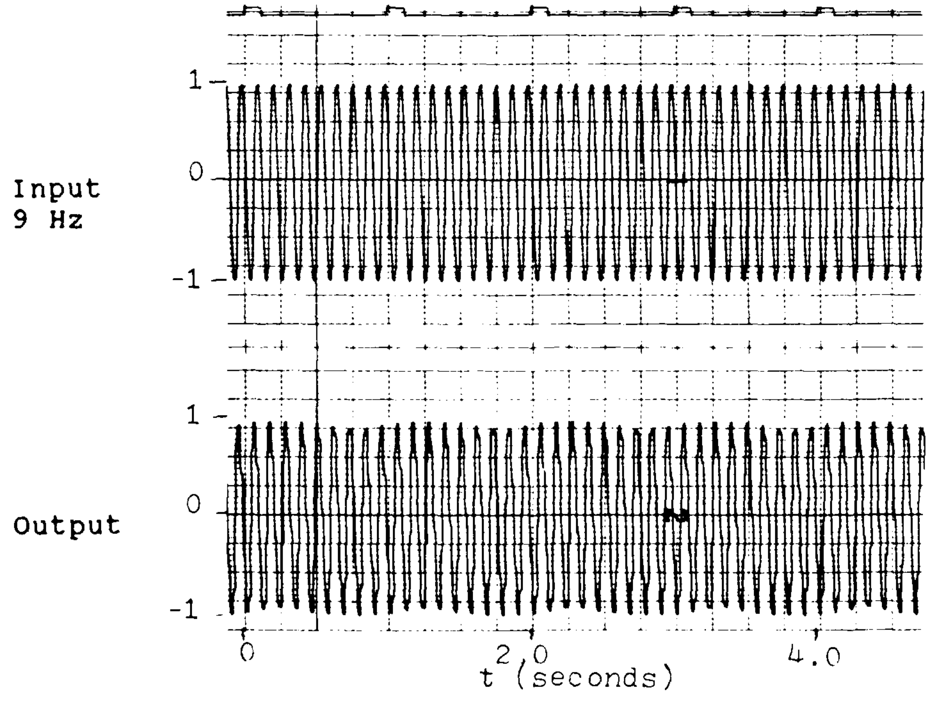

Figure 16. Subharmonic Aliasing Effect (input 9 Hz, sampled at 64 samples/sec)

$$c\ (t) = M*[A*\cos(\omega_o t)+B*\cos((\omega_s-\omega_o)t)] \qquad (13)$$

where M is the magnitude of the input signal prior to sampling. If M = 1 then A is the magnitude of this signal due to attenuation by the ZOH and B is the magnitude of the impostor frequency due to the non ideal filter characteristics of the ZOH.

Two constraints are imposed on the values of A and B. First,

$$A + B = 1 \qquad (14)$$

for all frequencies modeled. This constraint is imposed so that the magnitude M (Eq. 13) of the model is equal to the magnitude M (Eq 12) of the input signal for all frequencies modeled. The second constraint, which is also an assumption, is that when $\omega_o = \omega_s/2$, A must equal B. From the first constraint, this implies that for $\omega_o = \omega_s/2$

$$A = B = 0.5 \qquad (15)$$

The magnitudes of X from Figure 9 are

$$Y = X\ (\omega_o) = \frac{M * \sin\pi(\omega_o/\omega_s)}{\pi(\omega_o/\omega_s)} \qquad (16)$$

and

$$Z = X\ (\omega_s-\omega_o) = \frac{M * \sin\pi(\omega_s - \omega_o)/\omega_s}{\pi(\omega_s-\omega_o)/\omega_s} \qquad (17)$$

Assuming that A and B are similar sinc functions, where sinc(u) = [sin(u)]/u, define

$$A = \frac{\sin[\pi * x * (\omega_o/\omega_s)]}{\pi * x * (\omega_o/\omega_s)} \qquad (18)$$

$$B = \frac{\sin[\pi * x * (\omega_s - \omega_o/\omega_s]}{\pi * x * [(\omega_s - \omega_o)/\omega_s]} \qquad (19)$$

then for A=B=.5, and solving for x, gives $x \approx 1.2$

The plot of B, the magnitude of the impostor frequency, vs ω_o/ω_s is shown in Figure 17. The significance of this plot is evident when used in the following examples to verify the model and to determine the salient characteristics of the subharmonic aliased signal.

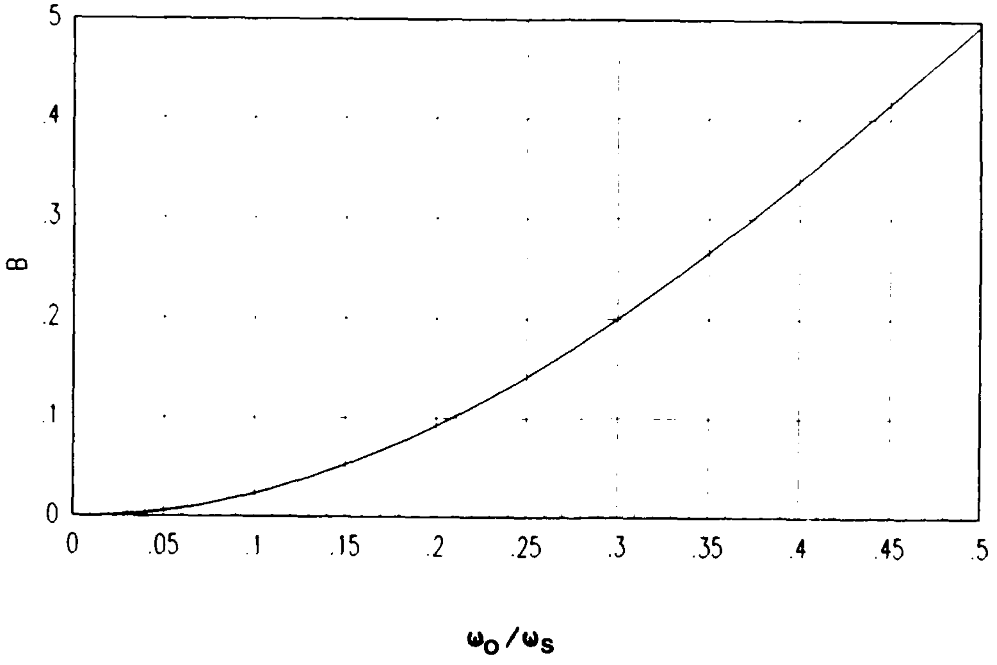

Figure 17. Magnitude of the Impostor Frequency (B) vs ω_o/ω_s for the Subharmonic Aliasing Model

To use Figure 17, a sample rate ω_s and input frequency (for a sine or cosine input) is selected and the ratio

ω_o/ω_s is found. Then, from Figure 17, B , the magnitude of the impostor frequency is found. Rearranging Eq. (14)

$$A = 1-B \tag{20}$$

the new magnitude of the input frequency, A, attenuated by the ZOH is calculated.

Model Examples

Example One. Let

$$\omega_s = 2*\pi*64 \tag{21}$$

$$\omega_o = 2*\pi*31 \tag{22}$$

$$T = 2*\pi/\omega_s = 0.015625$$

for an input

$$r(t) = M*\cos(\omega_o*t)$$

with M=1, the sampled output, from MATRIX , shown in Figure 18(a), is

$$c(kT) = \cos(2*\pi*31*k*T) \tag{23}$$

The model, from equation (13), is

$$c_m(t) = A*\cos(\omega_o t)+B*\cos(\omega_s-\omega_o)t$$

where

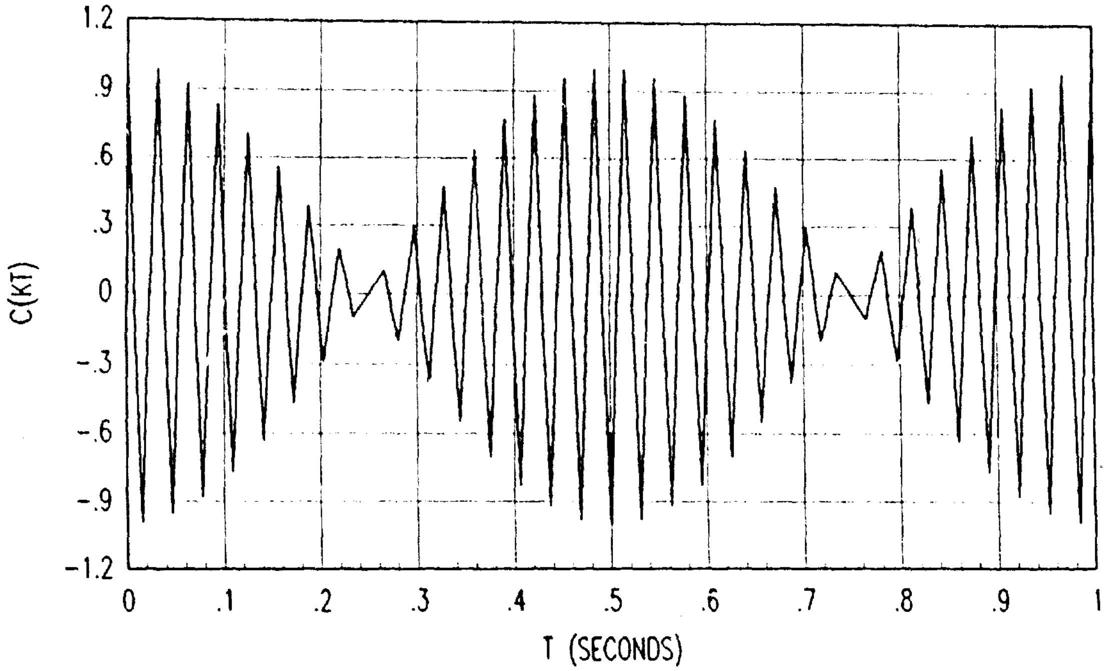

Figure 18(a). Output c(kt) = cos(2*π*31*k*T) Subharmonic Alias from Sampled Signal for f_o=31 Hz

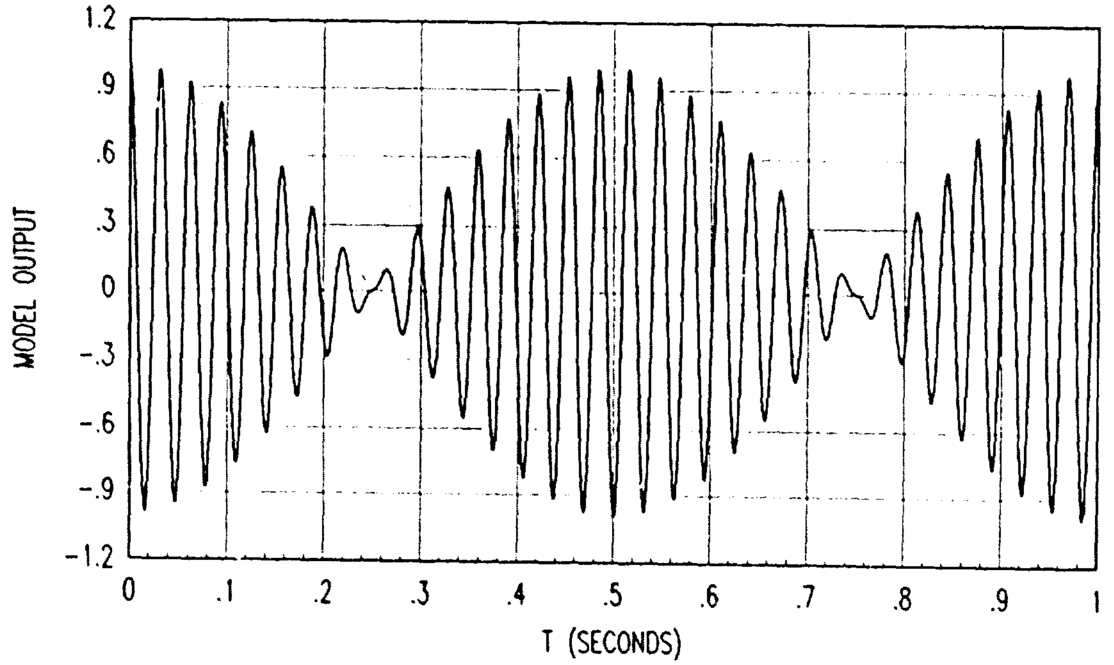

Figure 18(b). [illegible] *cos(2*π*31*t)+0.48*cos(2*π*33*t) [illegible] 'ng Model for f_o=31 Hz

$$\omega_s - \omega_o = 2*\pi*(64-31)$$

$$= 2*\pi*33$$

$$\omega_o/\omega_s = .484$$

From Figure 17, B=.48, so that A=1-.48=.52. The model is now defined as

$$c_m(t) = .52*\cos(2*\pi*31*t)+.48*\cos(2*\pi*33*t) \quad (25)$$

Figure 18(b) shows the model output.

Example Two. Let

$$\omega_s = 2*\pi*64 \quad (26)$$

$$\omega_o = 2*\pi*21 \quad (27)$$

$$T = 2*\pi/\omega_s = 0.015625$$

The sampled output, shown in Figures 19(a) and 20(a), is

$$c(kT) = \cos(2*\pi*21*k*T) \quad (29)$$

The model from equation (13) and Figure 17

for

$$\omega_s - \omega_o = 2*\pi*64 - 2*\pi*21 = 43$$

$$A = .77$$

$$B = .23$$

is now defined as

$$c_m(t) = .77*\cos(2*\pi*21*t)+.23*\cos(2*\pi*43*t) \quad (30)$$

Figures 19(b) and 20(b) show the model output.

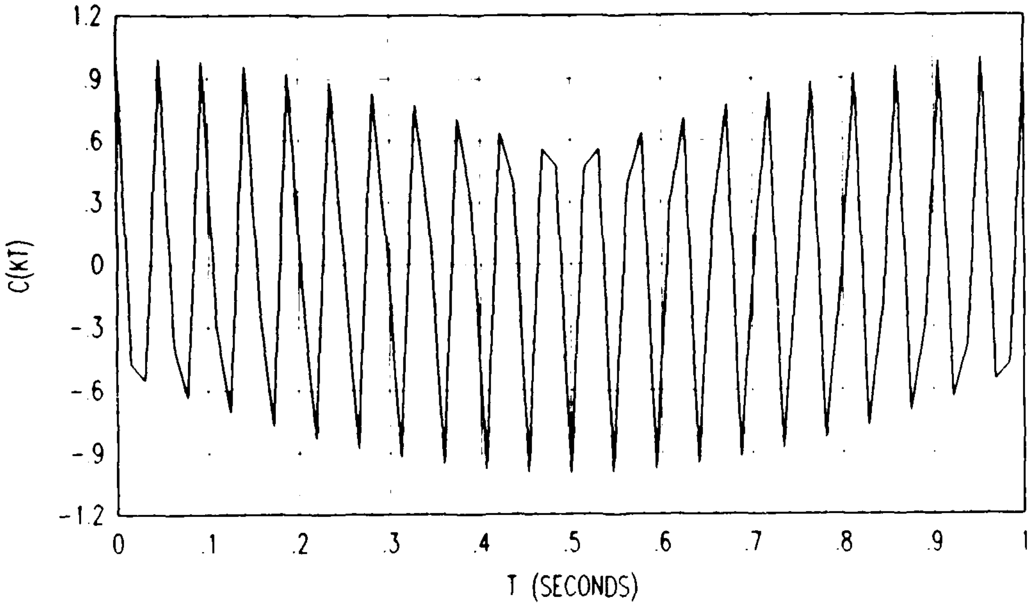

Figure 19(a). Output c(kt) = cos(2*π*21*k*T) Subharmonic Alias from Sampled Signal for f_o=21 Hz

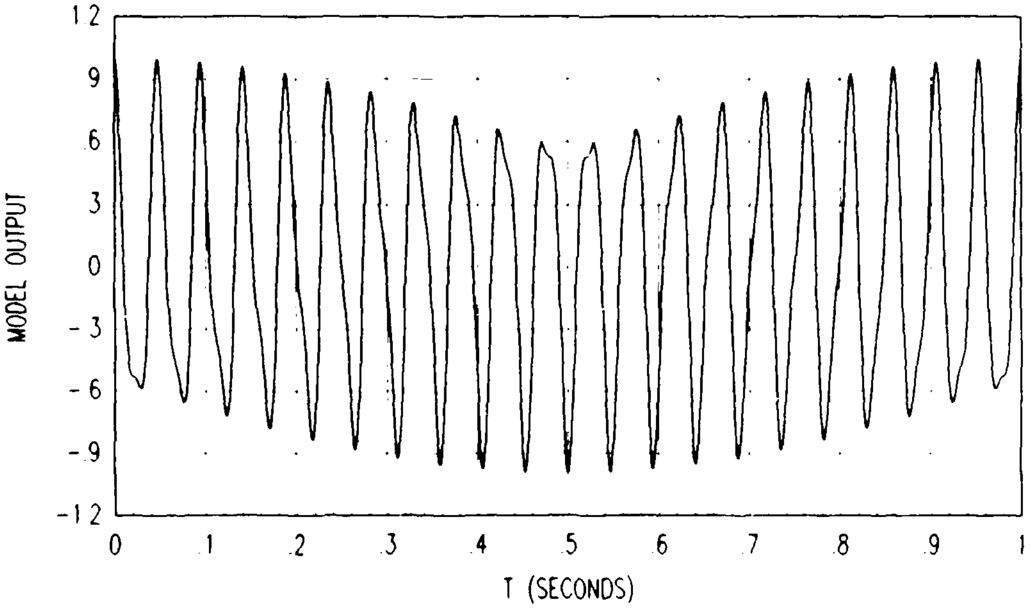

Figure 19(b). c(t)= 0.77*cos(2*π*21*t)+0.23*cos(2*π*43*t) Subharmonic Aliasing Model for f_o=21 Hz

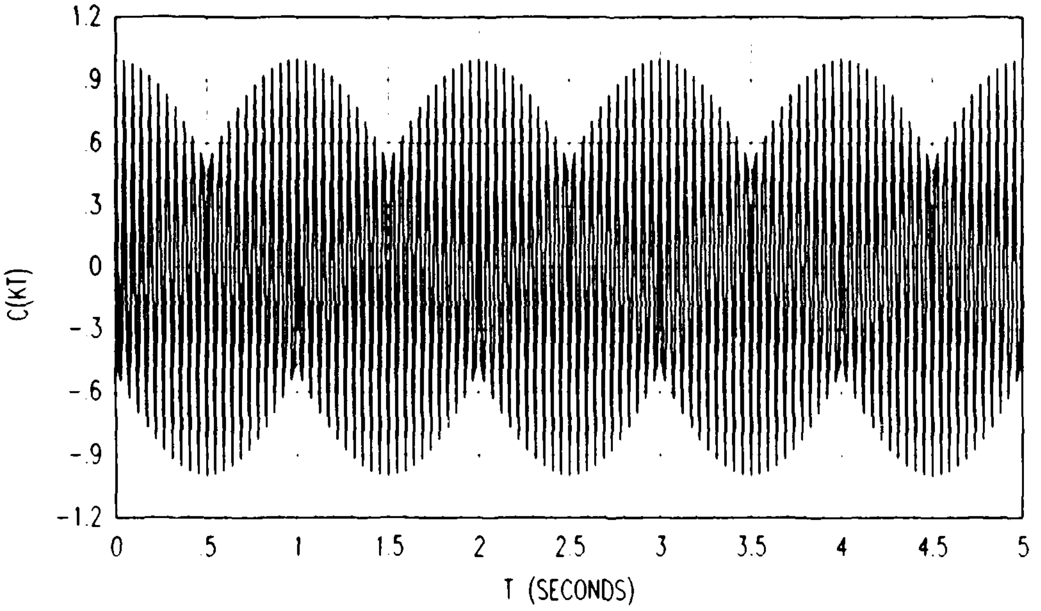

Figure 20(a). Output c(kt) = cos(2*π*21*k*T) Over several Subharmonic Alias Cycles for f_o=21 Hz

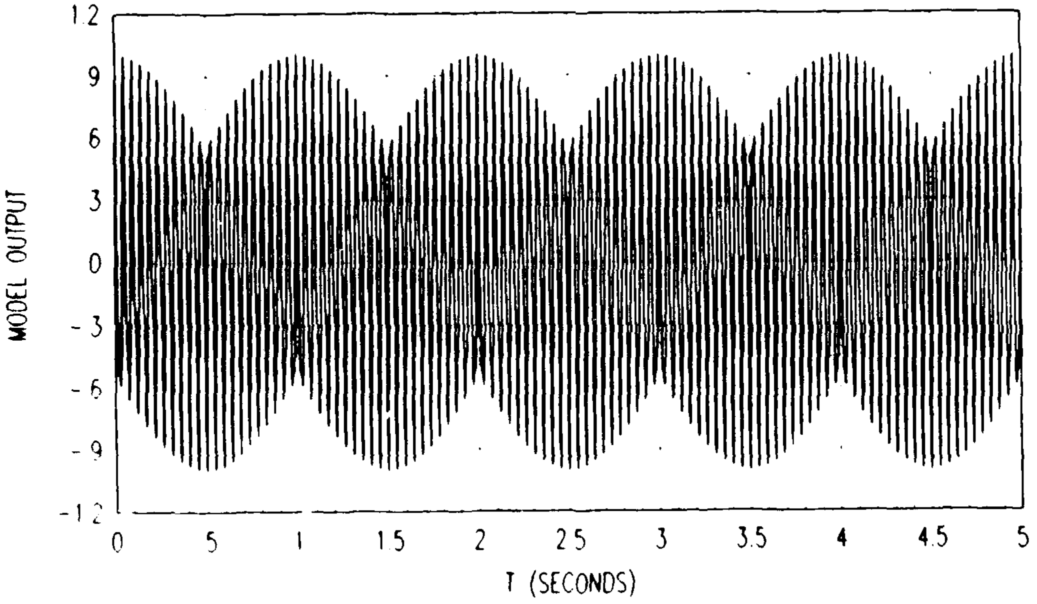

Figure 20(b). c(t)= 0.77*cos(2*π*21*t)+0.23*cos(2*π*43*t) Over Several Subharmonic Alias Cycles for f_o=31 Hz

Example Three. Let

$$\omega_s = 2*\pi*1 \tag{31}$$

$$\omega_o = 2*\pi*0.21 \tag{32}$$

$$M = 5$$

for an input

$$r(t) = 5*\cos(2*\pi*0.21*t)$$

The sampled output, shown in Figure 21(a), is given as

$$c(kT) = 5*\cos(2*\pi*0.21*k*T) \tag{33}$$

The model, from equation 13, is given by

$$c_m(t) = 5*[A*\cos(\omega_o t)+B*\cos(\omega_s-\omega_o)t]$$

where

$$\omega_s-\omega_o = 2*\pi*(1-0.21)$$

$$= 2*\pi*0.79$$

$$\omega_o/\omega_s = 0.21$$

From Figure 17, B=.1, so that A=1-.1=.9. The model is now defined as

$$c_m(t) = 5*[0.9*\cos(2*\pi*0.21*t) +.1*\cos(2*\pi*0.79*t)] \tag{34}$$

Figure 21(b) shows the model output.

Having validated the model, it can now be used to determine the frequency and magnitude of the subharmonic aliased signal. Two phenomena occur when two sinusoids are

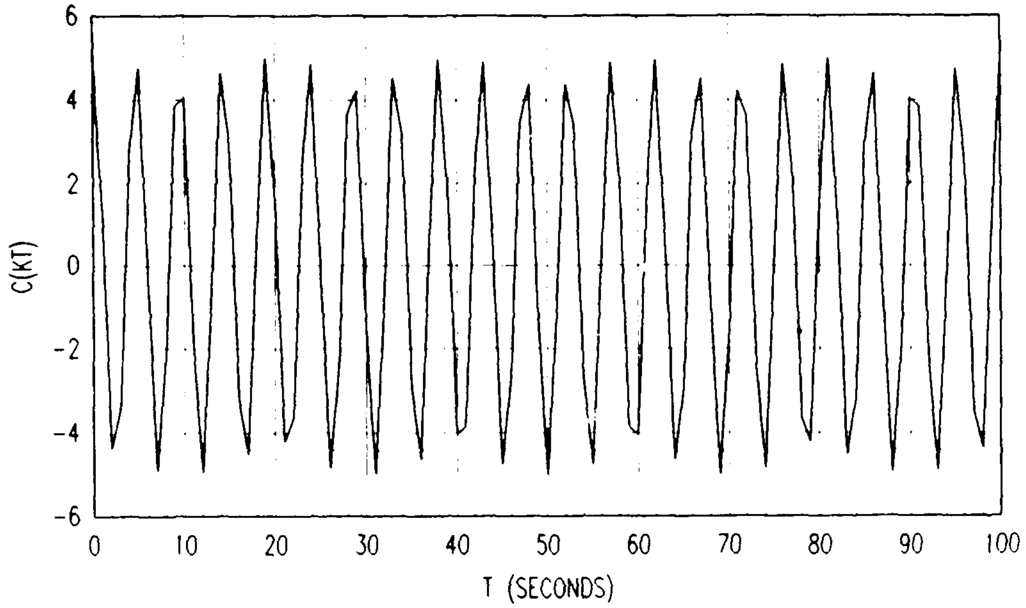

Figure 21(a). Output c(kt) = 5*cos(2*π*.21*k*T)
Subharmonic Alias from Sampled Signal for f_o=0.21 Hz

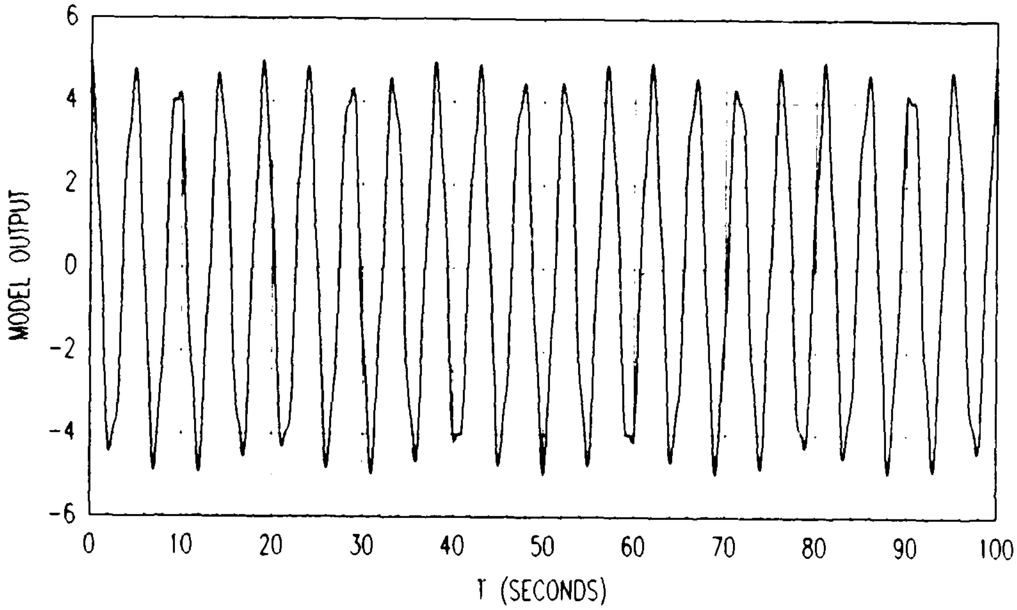

Figure 21(b). c(t)=5*[.9*cos(2*π*.21*t)+.1*cos(2*π*.79*t)]
Subharmonic Aliasing Model for f_o=0.21 Hz

added together, they are beating and apparent low frequency surge.

Beating occurs when the ratio of the two sinusoidal frequencies is nearly unity (13:21). Example 1, used previously, shows this phenomena well. The model frequencies are given as

$$\begin{aligned} f_o &= 31 \text{ Hz} \\ f_s - f_o &= 33 \text{ Hz} \end{aligned} \tag{36}$$

The beat frequency f_b is the absolute magnitude of the difference between these two frequencies.

$$f_b = 33-31 = 2 \text{ Hz} \tag{37}$$

The amplitude of the signal varies between the sum and the difference of the component amplitudes where

$$M*A = 1*0.52 = 0.52 \tag{38}$$

$$M*B = 1*0.48 = 0.48 \tag{39}$$

so that the amplitude varies between

$$0.52 + 0.48 = 1 \tag{40}$$

and

$$0.52 - 0.48 = 0.04 \tag{41}$$

So, the resultant wave has the same apparent frequency as the major component (ie. that with the greater amplitude) (13:21) of 31 Hz, its amplitude varies between 1.0 and .04, and it has a beat frequency of 2 Hz. This is shown in Figure 22.

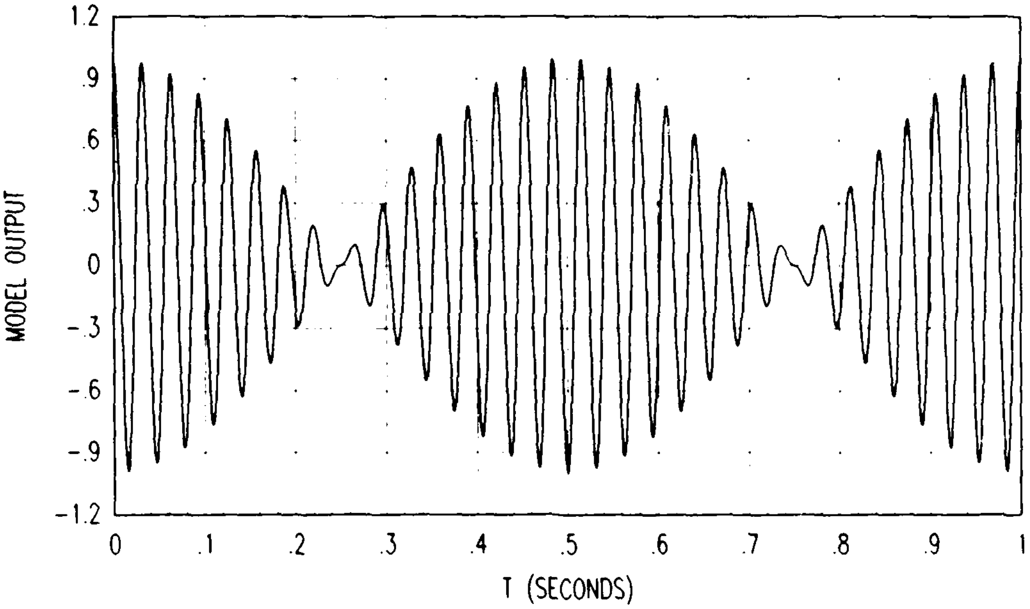

Figure 22. c(t)= 0.52*cos(2*π*31*t)+0.48*cos(2*π*33*t) Subharmonic Alias with a Beat Frequency for f_o=31 Hz

An apparent low-frequency surge occurs when the ratio of the two sinusoidal frequencies is nearly an interger (n) (when the larger frequency is divided by the smaller frequency) (13:21). Example 2, used previously, shows this phenomena well. The model frequencies are

$$f_o = 21 \text{ Hz} \tag{42}$$

$$f_s - f_o = 43 \text{ Hz} \tag{43}$$

The apparent low-frequency surge f_{al} is the absolute magnitude of the difference between $n*f_o$ and $f_s - f_o$ (13:39), where n, in this case is equal to two. Therefore, the apparent low-frequency surge is

$$f_{al} = 43-(2*21) \quad (44)$$
$$= 1 \text{ Hz}$$

Again, the amplitude of the signal varies between the sum and the difference of the component amplitudes where

$$M*A = 1*0.77 = 0.77 \quad (45)$$

$$M*B = 1*0.23 = 0.23 \quad (46)$$

so that the amplitude varies between

$$0.77 + 0.23 = 1 \quad (47)$$

and

$$0.77 - 0.23 = 0.54 \quad (48)$$

So, the resultant wave has the same apparent frequency as the major component (ie. that with the greater amplitude) of 21 Hz, its amplitude varies between 1.0 and .54, and it has an apparent low-frequency surge of 1 Hz. This is shown in Figure 23.

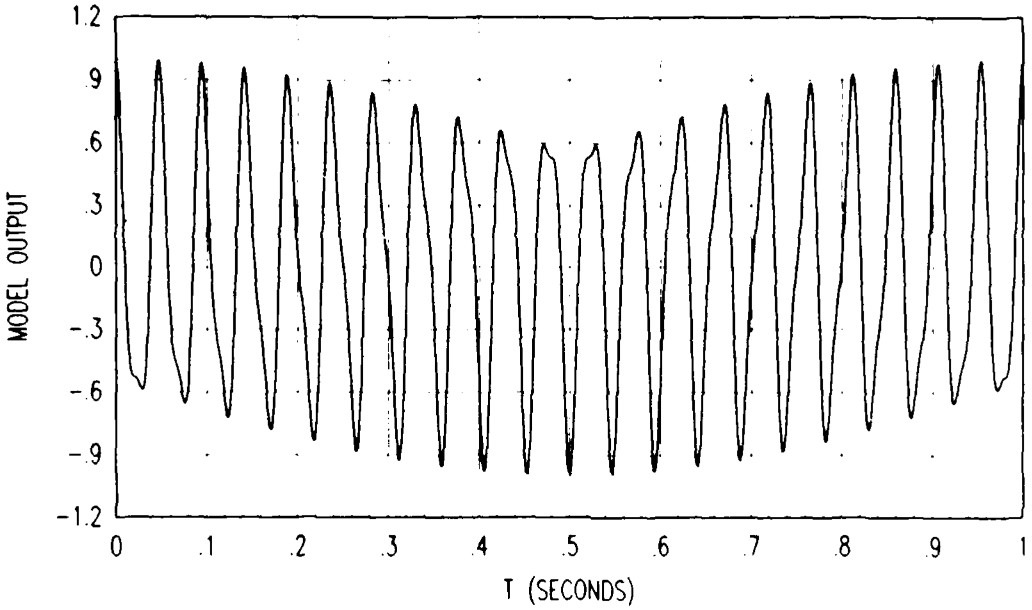

Figure 23. Subharmonic Aliasing with an Apparent Low Frequency Surge

Utility of the Model

Now that the model is well defined, how can it be used? One possible application is to use the subharmonic alias model as an input (known to be within the subharmonic range, $\omega_s/10 < \omega_o < \omega_s/2$), so that the output characteristics of the system can be predicted. A simple system with a first order filter is used to demonstrate this.

First Order Filter Example. The purpose of this example is to show that the subharmonic aliasing model can be used much like the pseudo continuous time model (8:248) to predict the output characteristics of a digital system.

First, the system in Figure 24(a) is simulated on a hybrid computer to determine the base-line system response, that is, the actual response of the anolog/digital control system. The system parameters are

$$r(t) = \cos(2*\pi*31*t) \text{ volts}$$

$$T = 0.015625 \text{ seconds}$$

$$\omega_s = 2*\pi/T = 2*\pi*64 \text{ (radians/second)}$$

where $\omega_o = 2*\pi*31$ of r(t) is within the subharmonic range, $2*\pi*6.4 < \omega_o < 2*\pi*32$, for this example. The steady state output response for this system is found by determining the frequency response of the first order filter. For the input frequency $\omega_o = 2*\pi*31$ the output is attenuated by ≈ -7 dB with a phase shift $\phi = -63°$. Therefore, the output equation for this system is

$$C(kT) = .45*\cos(2*\pi*31*t - 63°) \qquad (49)$$

This output, shown in Figure 25 and labeled C(kT), is the hybrid computer response to the system modeled in Figure 24(a) and correlates well with output equation (49). Unfortunately, equation (49) says very little about the subharmonic characteristics of the output in Figure 25 which definitely shows a beat frequency of 2 Hz with an amplitude varying from approximately 0.02 to o.45 volts.

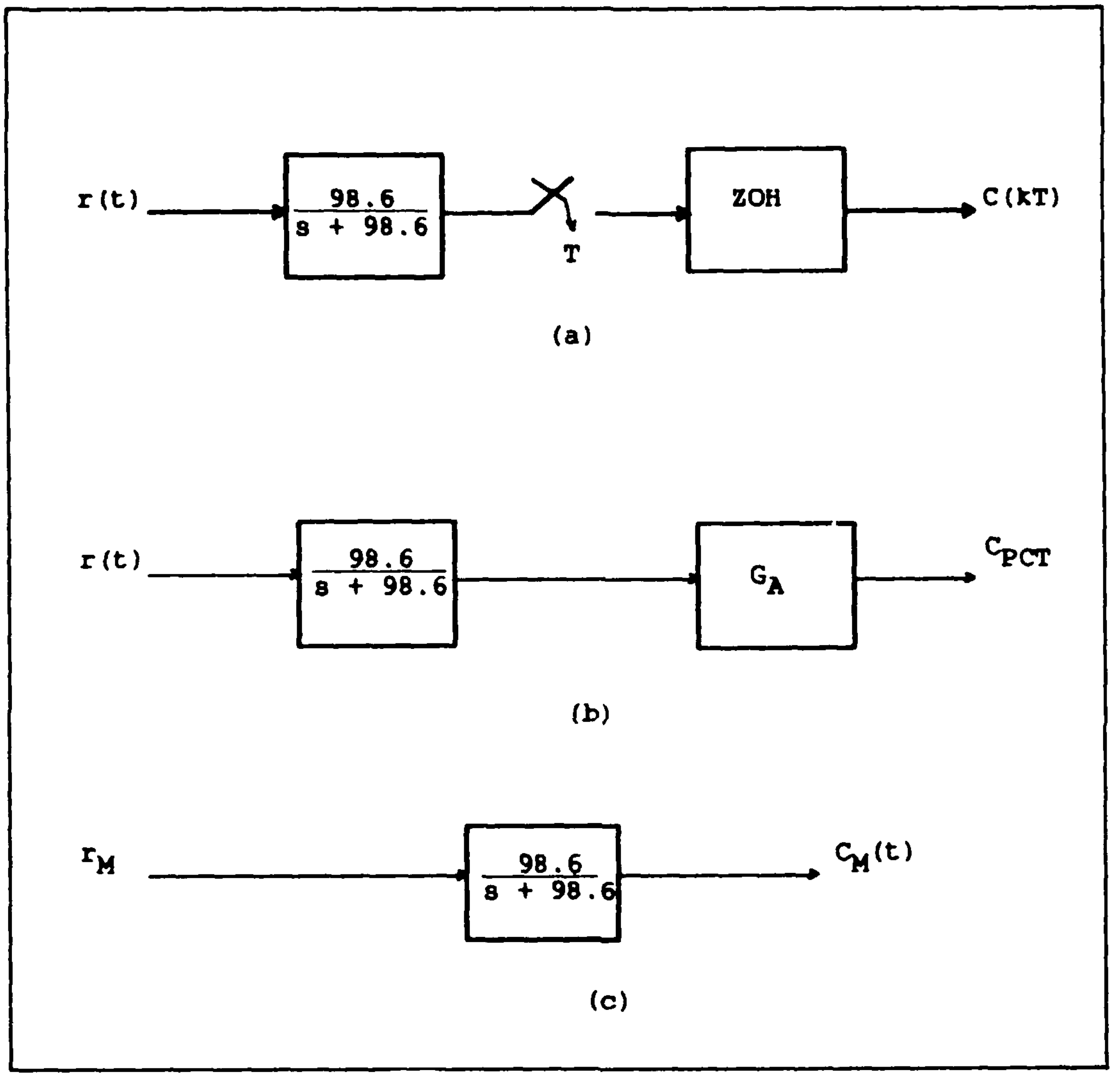

Figure 24. System Models

Second, the system is modeled using the pseudo continuous time model shown in Figure 24(b). The sampler and the zero order hold of the digital system are

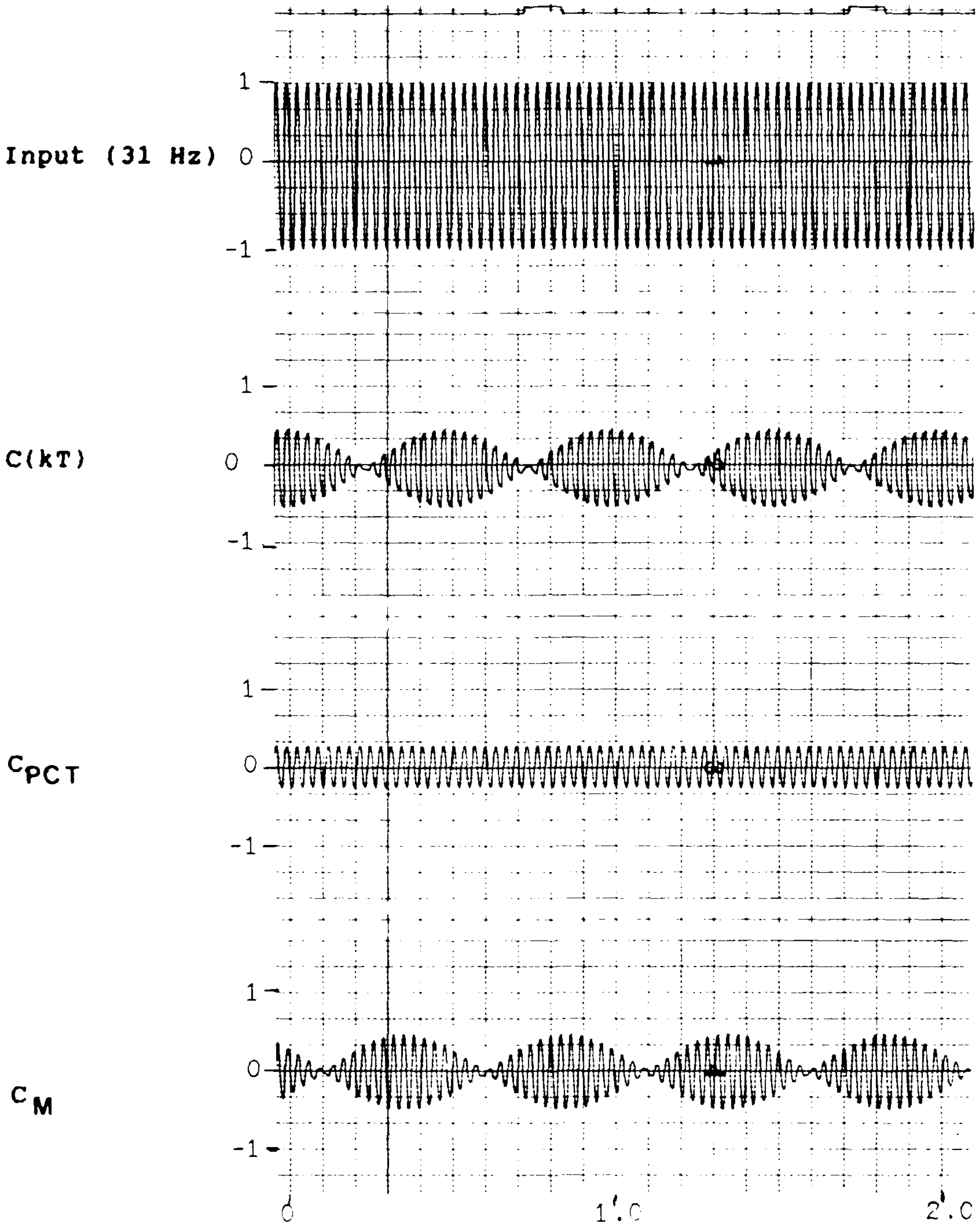

Figure 25. Outputs from Hybrid Simulation C(kT), Pseudo Continuous Time Simulation C_{PCT}, and Model Simulation C_M

approximated by using 1/T and the first order Pade' approximation (8:249) respectively to form G_A

$$G_A = \frac{2/T}{s + 2/T} \qquad (50)$$

Substituting the sample period T=0.015625 seconds into (50)

$$G_A = \frac{128}{s+128} \qquad (51)$$

The steady state system output response C_{PCT} for the pseudo continuous time second order system model is

$$C_{PCT} = 0.249*\cos(2*\pi*31*t - 120^\circ) \qquad (52)$$

This output response is shown in Figure 25 and is labeled C_{PCT}. This is not a good model of this system for two reasons. First, it does not reflect the existence of a subharmonic alias, which is known to exist, as shown in output C(kt). Second, the model itself lies outside the "good" Tustin region(8:249, which is described as the s-domain region which adequately models a discrete system in the continuous domain so that continuous control law can be used to determine the system response.

Third, model the input using the subharmonic aliasing aliasing model

$$r_M(t) = A*\cos(\omega_o t) + B*\cos(\omega_s - \omega_o)t \qquad (53)$$

where

$\omega_o = 2*\pi*31$

$\omega_s = 2*\pi*64$

$\omega_o/\omega_s = 0.484$

so from Figure 17

$B = 0.48$

and $A = 1-B = 0.52$

The input r(t) is now defined as

$$r(t) = 0.52*\cos(2*\pi*31*t) + 0.48*\cos(2*\pi*33*t) \qquad (54)$$

For linear systems (which this is) the principle of superposition can be used to calculate the output. The two components of the modelled input can be considered one at a time. The output signal is then equal to the sum of the contributions produced by each input (15:162). Let C_{M1} be the output response to 0.52*cos(2*π*31*t) and C_{M2} be the output response to 0.48*cos(2*π*33*t). Therefore, the steady state responses are

$$C_{M1} = .235*\cos(2*\pi*31*t - 63^\circ) \qquad (55)$$

$$C_{M2} = .206*\cos(2*\pi*33*t - 64^\circ) \qquad (56)$$

Combining (55) and (56) the output CM becomes

$$C_M(t) = 0.235*\cos(2*\pi*31*t - 63^\circ) + 0.206*\cos(2*\pi*33*t - 64^\circ) \qquad (57)$$

This output, shown in Figure 25 and labeled as $C_M(t)$ shows a definite subharmonic alias whose characteristics can now be calculated. From (57) the beat frequency can be calculated as (33-31) Hz = 2 Hz. The amplitude of the beat frequency varies between 0.441 and 0.029.

This example demonstrates that the subharmonic aliasing input model produces the salient characteristics of the digital (or hybrid) subharmonic aliased output. The model output can then be used to predict the subharmonic aliasing characteristics.

V. Subharmonic Aliasing and the Interchannel Difference

The purpose of this chapter is to provides a means to predict the interchannel difference for input frequencies within the subharmonic range. In Chapter II, the maximum sample period is calculated with Equation (1) so that the interchannel difference would be less than five percent

$$T = e_{max}/\dot{s} \qquad (1)$$

where T is the maximum sample period, e_{max} is the maximum interchannel difference, and $\dot{s}$ is the maximum signal rate of change. As described in the Data Aquisition section of Chapter II, the sensor sample rate T for the AFTI/F-16 is 0.00390625 seconds. Since the sensor sample rate (1/T) is fixed, the maximum interchannel difference for an input can easily be calculated by rearranging Equation (1) to

$$e_{max} = T * \dot{s} \qquad (58)$$

Given a sinusoidal input

$$r(t) = M*\cos(\omega_o*t) \qquad (59)$$

the maximum rate of change $\dot{s}$ is found by differentiating r(t) with respect to time

$$\dot{s} = \max_{\omega_o t}\{dr(t)/dt\} \qquad (60)$$

$$\dot{s} = \max_{\omega_o t}\{M*\omega_o*\sin(\omega_o*t)\}$$

$$= \pm M * \omega_o$$

for $\omega_o t = \pm n\pi/2 \quad n=1,3,5,...$

Substituting Equation (59) into Equation (58) for n=1 gives

$$e_{max} = T * M *\omega_o \tag{61}$$

Let e be defined as the maximum interchannel difference when M=1, therefore,

$$e = T * \omega_o \tag{62}$$

Substituting (62) into (61)

$$e_{max} = M * e \tag{63}$$

A graph of Equation (62) shows the relationship between the frequency of an input signal (ω_o) and the maximum error e_{max} (normalized) caused by time skew between computers for the sensor sample period of 3.90625 msec (Figure 26). From this figure, it is obvious that the high frequencies (ie. frequencies in the subharmonic range) cause large interchannel differences, which may exceed the Input Selector Monitor trip levels. If this frequency (input signal) persists sufficiently long, a sensor may be voted as failed and removed from the failure management scheme.

Example

The inter-branch trip level is 15% of full scale for most AFTI/F-16 inputs. This example uses a hypothetical input of -4.5 to 4.5 volts full scale and a trip level (TL)

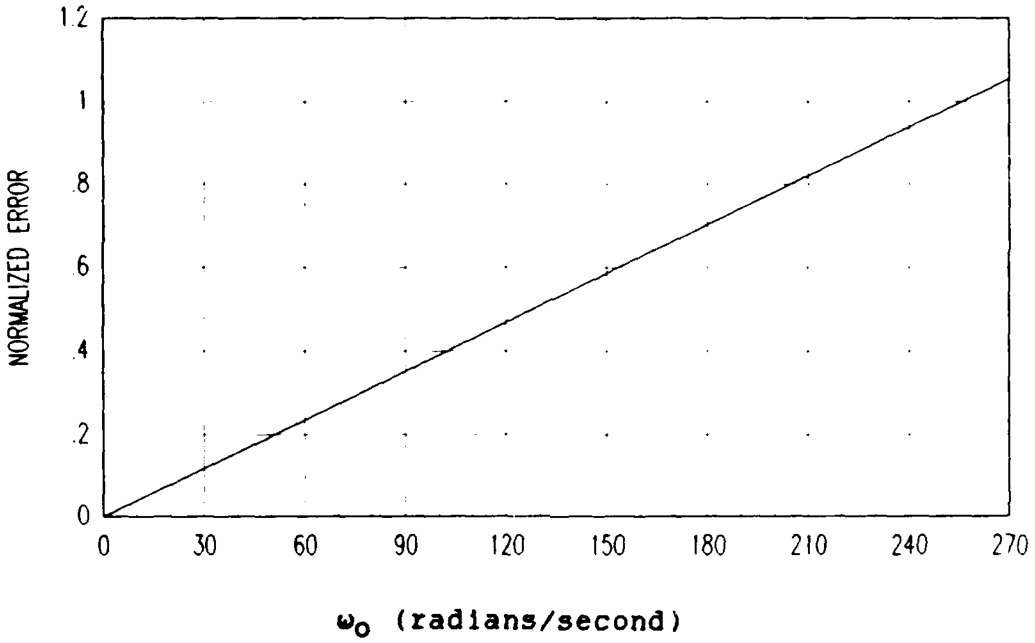

Figure 26. Maximum Interchannel Difference for Maximum Time Skew(.00390625 seconds)

of ± 15 percent, so that the trip level for this example is

$$\begin{aligned} TL &= \pm 9 * 0.15 \\ &= \pm 1.35 \end{aligned} \tag{64}$$

Given an input

$$r(t) = 3*\cos(2*\pi*21.186*t) \text{ (volts)}$$

The maximum rate of change for this input is

$$\dot{s} = 3*2*\pi 21.186 \text{ (volts)}$$

The sensor sample period T is 0.00390625 seconds, therefore the maximum interchannel difference from Equation (58) is

$$e_{max} = \pm 0.00390625*3*2*\pi*21.1816$$
$$\approx \pm 1.56 \text{ volts}$$

which is greater than the trip level of ± 1.35 volts. This implies that for this input magnitude (M = 3) and frequency (ω_o = 21.186 Hz) if two FLCC's are time skewed by the maximum amount (0.00390625 seconds) a transient error occurs at least twice per cycle. This example is simulated on a digital computer MATRIX and a hybrid computer (SIMSTAR Programs, S1.SATL,Appendix B) for maximum time skew with results shown in Figure 27 and Table III. Figure 27(a) is the time response of the input r(t) = 3*cos(2*π*21.186). Figure 27(b) is the time response of the interchannel difference for maximum time skew. This figure is generated by sampling the input r(t) every 0.00390625 seconds and subtracting the most recent past sampled value by the present sampled value. Figure 27(b) clearly demonstrates that the trip level (drawn in at ± 1.35) will be exceeded at least eight times within the first 0.1 seconds for this input. The data items marked by three asterisks in Table III, show that there are often more than two trips per input cycle. This chapter demonstrates the sensitivity of the interchannel difference with respect to frequencies in the subharmonic range and the ease of calculating the maximum interchannel difference e_{max} for sinusoidal inputs.

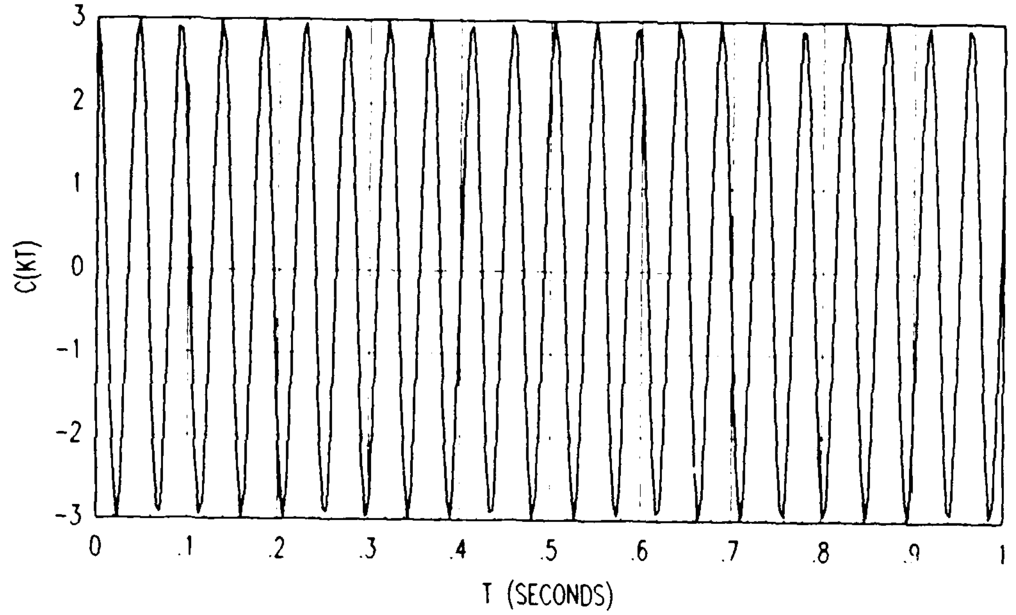

Figure 27(a). Input Frequency (21.186 Hz) Sampled every 0.00390625 sec. (the Sensor Sample Period for the DFCS)

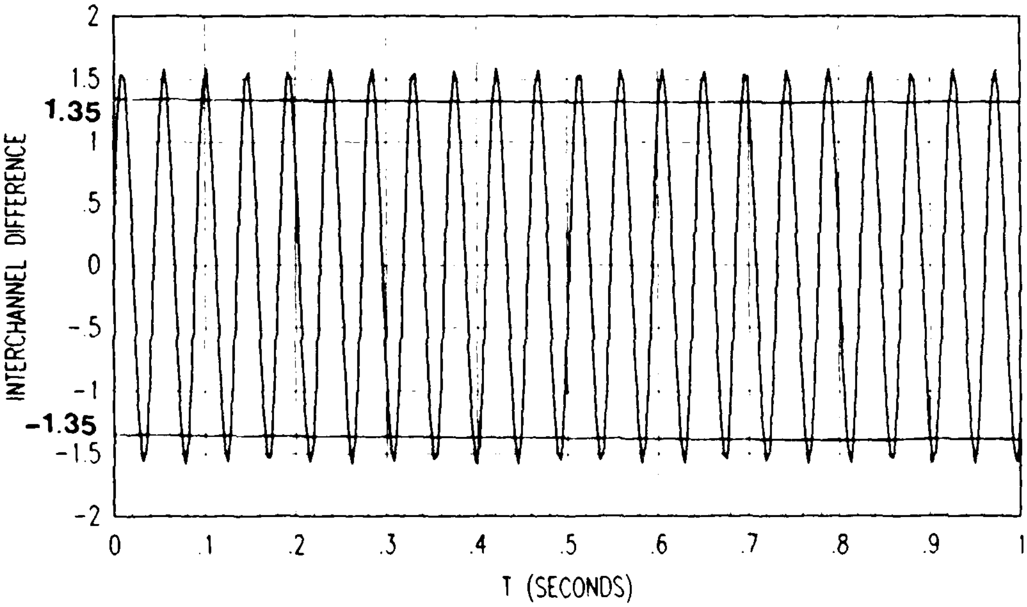

Figure 27(b). The Interchannel Difference for Maximum Time Skew (0.00390625 sec.)

Table III. Interchannel Difference for Maximum Time Skew
(Input r(t)=3*cos(2*π*21.186) sampled every.00390625 sec.)

LINE	TIME	SAMPLED	INTERCHANNEL	
		INPUT	DIFFERENCE	
0	0.0	0.00000	0.0	
1	0.003906	2.20647	-2.20647	***
2	0.007813	2.9983	-0.791831	
3	0.011719	2.13425	0.864045	
4	0.015625	0.094183	2.04007	***
5	0.019513	-1.99803	2.09221	***
6	0.023438	-2.98926	0.991235	
7	0.027344	-2.32777	-0.661489	
8	0.03125	-0.3062	-2.02157	***
9	0.035156	1.78601	-2.09221	***
10	0.039063	2.95152	-1.16551	
11	0.042969	2.50258	0.448946	
12	0.046875	0.672736	1.82984	***
13	0.050781	-1.52711	2.19985	***
14	0.054688	-2.88436	1.35724	***
15	0.058594	-2.65068	-0.233671	

VI. Rate Limiter

The purpose of this chapter is to demonstrate the rate limiting effect that the AFTI/F-16 engineers described as "the rate limiter following the low frequency alias waveforms, instead of the input waveform".(1:10) Also, for the benefit of AFTI/F-16 engineers, this chapter provides the output for comparison of two different rate limiter configurations: the inputs averaged then rate limited, the inputs rate limited then averaged. Hybrid computer simulations are used to provide the outputs.

Chapter IV shows how the analog to digital and digital to analog processes induce impostor frequencies which distort the original input. At predetermined frequencies (based on the sample rate) well defined subharmonic aliases are formed, with beat frequencies or apparent low frequency surges. Chapter V verifies that inputs in the subharmonic range $\omega_s/10 < \omega_o < \omega_s/2$ have a high rate of change, which effects the magnitude of the interchannel difference.

The purpose of the digital flight control system software rate limiter is to limit this rate of change to no more than five percent full scale per computational frame. The combination of three elements:

1. the apparent low frequency surge,
2. the input high rate of change and,

3. the software rate limiters nonlinear characteristics produce the effect of the rate limiter following the "low frequency component".

Since the DFCS control laws and aircraft dynamics are not simulated, a generic input to the rate limiter is used to simulate the effect of the rate limiter. To determine a reasonable rate limit, the example from Chapter V is used again. Full scale is defined as -4.5 to 4.5 volts. To limit this to five percent per computation frame the rate limit threshold (T_{RL})

$$T_{RL} = 9 \text{ volts} * 0.05$$
$$= 0.45 \text{ volts}$$

The pseudo-code for rate limiter is provided below and is also shown in block diagram form in Figure 28.

```
if old_output - new_input > rate_limit_threshold  then
        new_output = old_output - rate_limit_threshold
else if old_output - new_input < - rate_limit_threshold then
        new_output = old_output + rate_limit_threshold
else    new_output = new_input
end if
        old_output = new_output
```

where old_output, (Shown in Figure 28 as O[(k-1)*T)]) is the rate limiters previously computed output, new_output {O(k*T)} is the the output of the rate limiter, new_input

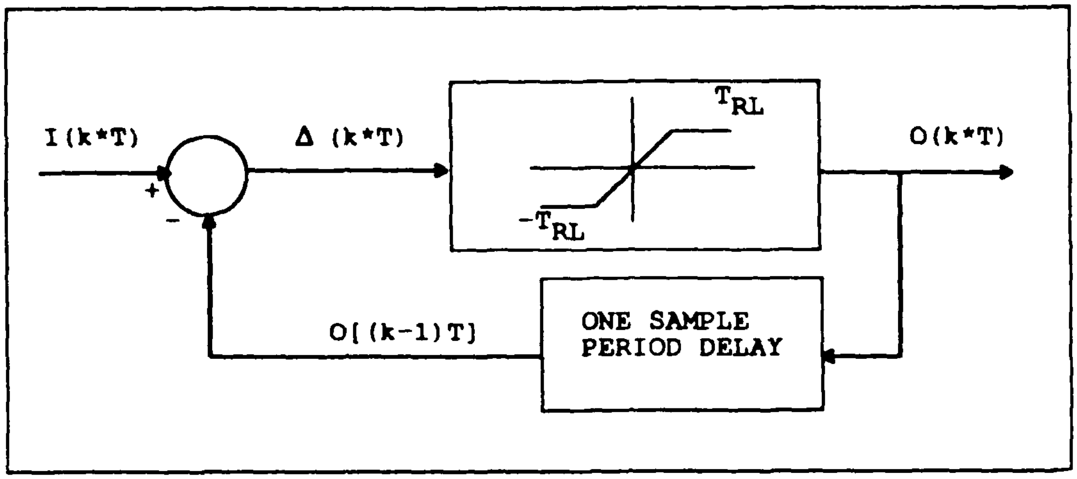

Figure 28. Software Rate Limiter Block Diagram

{I(k*T)} is the input to the rate limiter, Δ(k*T) is the difference between old_output and new_input, and T_{RL} is the rate_limit_threshold. This pseudo-code is similar to the FORTRAN code used in the hybrid simulations.

Three software rate limiting simulations are performed on a hybrid computer (SIMSTAR Programs, S1.RLSIM, Appendix B) to demonstrate the rate limiting effect on the subharmonic alias. The first simulation (Block Diagram, Figure 29(a)) rate limits a single input simply to show the rate limiter following the "low frequency" of the subharmonic aliased input). The second simulation (Figure 29(b)) provids two inputs, time skewed by one sample period (0.015625 second, which is approximately one FLCC frame) which are averaged, then input to the rate limiter. The third simulation (Figure 29(c)) rate limited the two time skewed signals and then averaged them.

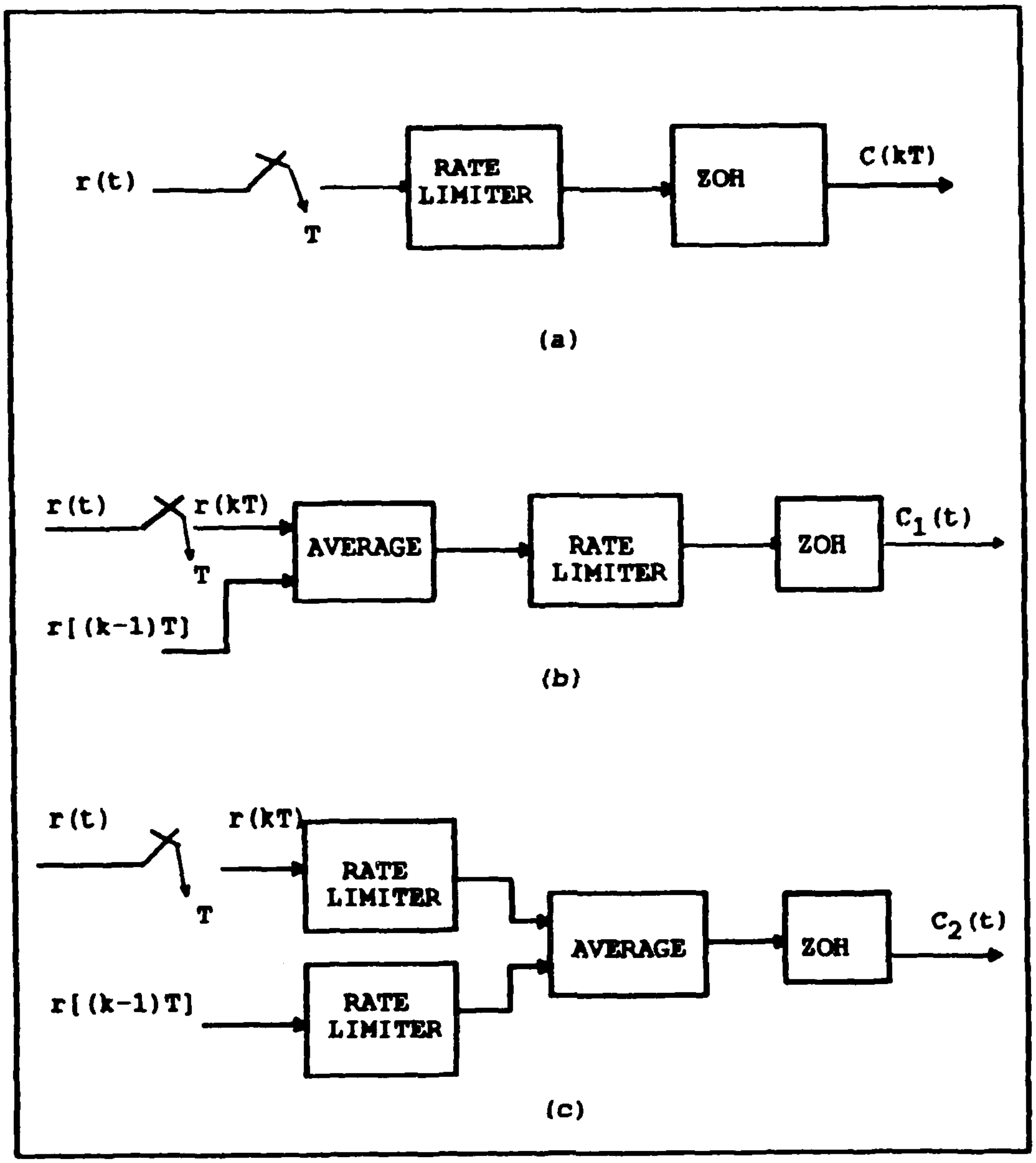

Figure 29. Block Diagrams for Rate Limiter Simulations
(a) Rate Limiter and Subharmonic Alias Simulation
(b) Dual Input Average to Rate Limiter Simulation
(c) Dual Input Rate Limiter to Average Simulation

Rate Limiter Example

To illustrate the effects of the rate limiter, a low frequency is sampled at a low sample rate to induce subharmonic aliases and, at the same time, minimize the strip chart recorder pin dynamics. Figure 30 shows the input, D/A output, and the rate limiter response. The input

$$r(t) = \cos(2*\pi*f_o*t)$$

where f_o = 1 Hz sampled at 3.0476 samples/sec. From Chapter 4, this input and sample rate produce a well defined subharmonic alias with an apparent low frequency surge of

$$\begin{aligned} f_o &= (3*1) - 3.0476 \\ &= .0476 \text{ Hz} \end{aligned}$$

The rate limit threshold is set at 0.45 volts per computation frame, which, for a step input with magnitude 2 to the rate limiter requires 5 samples for the output of the rate limiter to reach 2. This is calculate by dividing the magnitude of the step input by the rate limit threshold and rounding up to the nearest integer, for example, 2/0.45 ≈ 4.44, rounding up = 5. With the aid of the rate limiter pseudo-code and points labeled on Figure 34 the effect of rate limit is described. In general, for a rate limit threshold of T_{RL} = .45, comparing old_outputs (A' - E' in Figure 30) with new_inputs (B - E in Figure 30) the

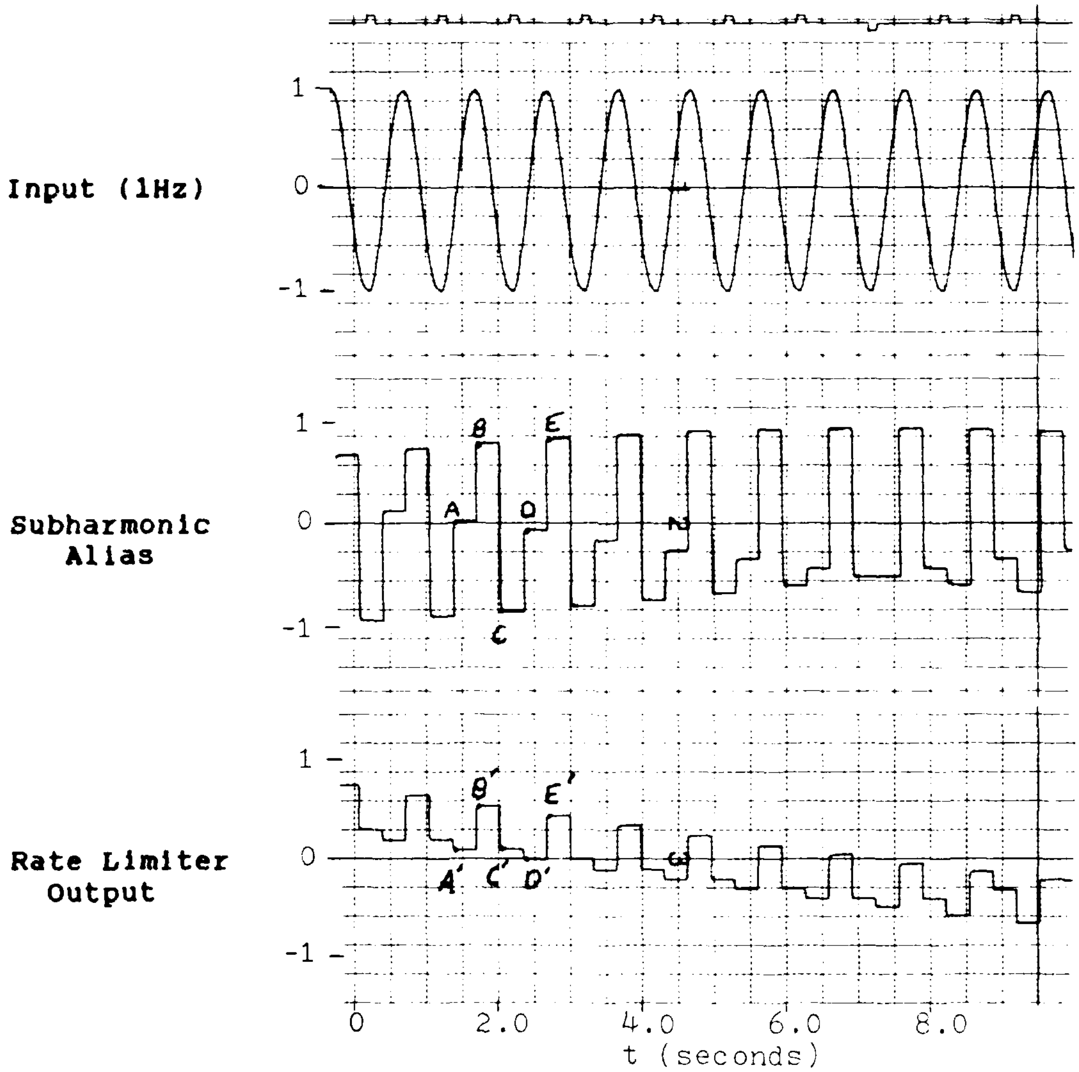

Figure 30. Rate Limiter Output to 1 Hz Input
Sampled at 3.0476 samples/second
(A, B, C, D, and E are values of the subharmonic alias)
(A',B', C', D', are E' are rate limiter output values)

new_outputs can be calculated. For this example A' - B is less than the rate_limit_threshold

$$A' - B < -T$$

so that the new_output is

$$B' = A' + T$$

Similarly,

$$B' - C > T$$

so that the new_output is

$$C' = B' - T$$

and

$$C' - D < T$$

so that the new_output is equal to the new_input

$$D' = D$$

These three rate limits are examples of the three states of the rate limiter with

	$A' - B' < -T$	which exceeds the rate limit
	$B' - C > T$	which exceeds the rate limit
and	$C' - D < T$	which is within the rate limit

Figure 31 is this example run on a hybrid computer (SIMSTAR) with a slower strip chart recorder speed to show two full cycles of both the subharmonic alias and the rate limiter output. This is an good example of how the non-linear characteristics of the software rate limiter, and the characteristics of the subharmonic alias combine so that the rate limiter follows the "low frequency".

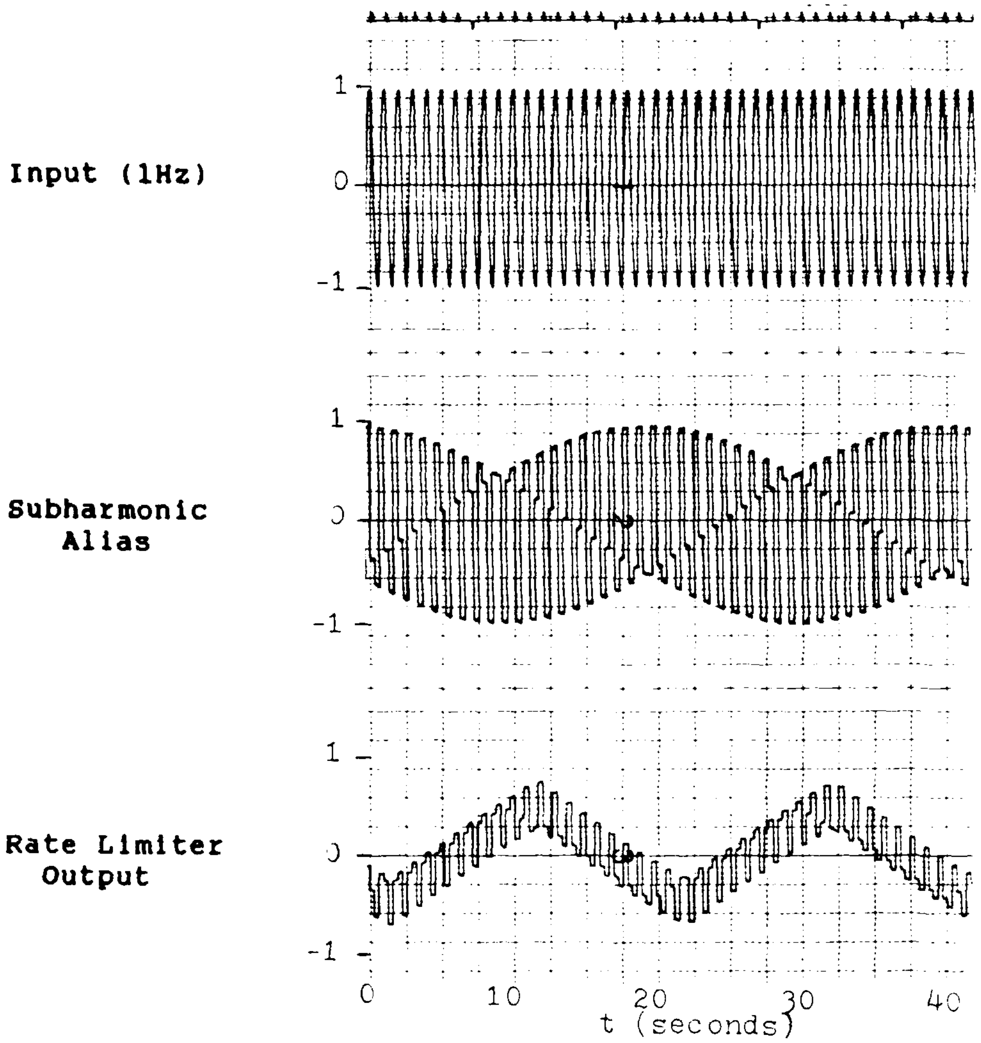

Figure 31. Rate Limiter Output to 1 Hz Input Sampled at 3.0476 samples/second

Comparison of Techniques

This section compares the outputs of the two different possible rate limiting techniques: averaging the inputs and then rate limiting (C , Figure 29(b)), limiting the inputs and then averaging (C , Figure 29(c)). Inputs are of the form $r(t) = \cos(\omega_O*t)$, sampled at 64 samples/second with with a rate limit threshold T_{RL} = 0.45.

For ω_O =2*π*31 (Figure 32), both techniques effectively rate limit the input signal. For ω_O =2*π*21 (Figure33), rate limiting the inputs prior to averaging better limits the high frequency, but the magnitude of the low frequency is greater than that of the configuration which averages the inputs prior to rate limiting. For ω_O =2*π*17 (Figure 34), rate limiting prior to averaging decreases the magnitude of both the high and low frequencies better than averaging and then rate limiting. For ω_O =2*π*13 Hz, another well defined subharmonic alias (Figure 35), rate limiting prior to averaging, once agian, better limits the input signals. Finally, at .5 Hz (Figure 36), (which is near the short period mode of the AFTI/F-16) there is no limiting for this input - which is expected considering the low rate of change of the input.

In general, rate limiting the inputs prior to averaging limits the input signal better than averaging the input and then rate limiting.

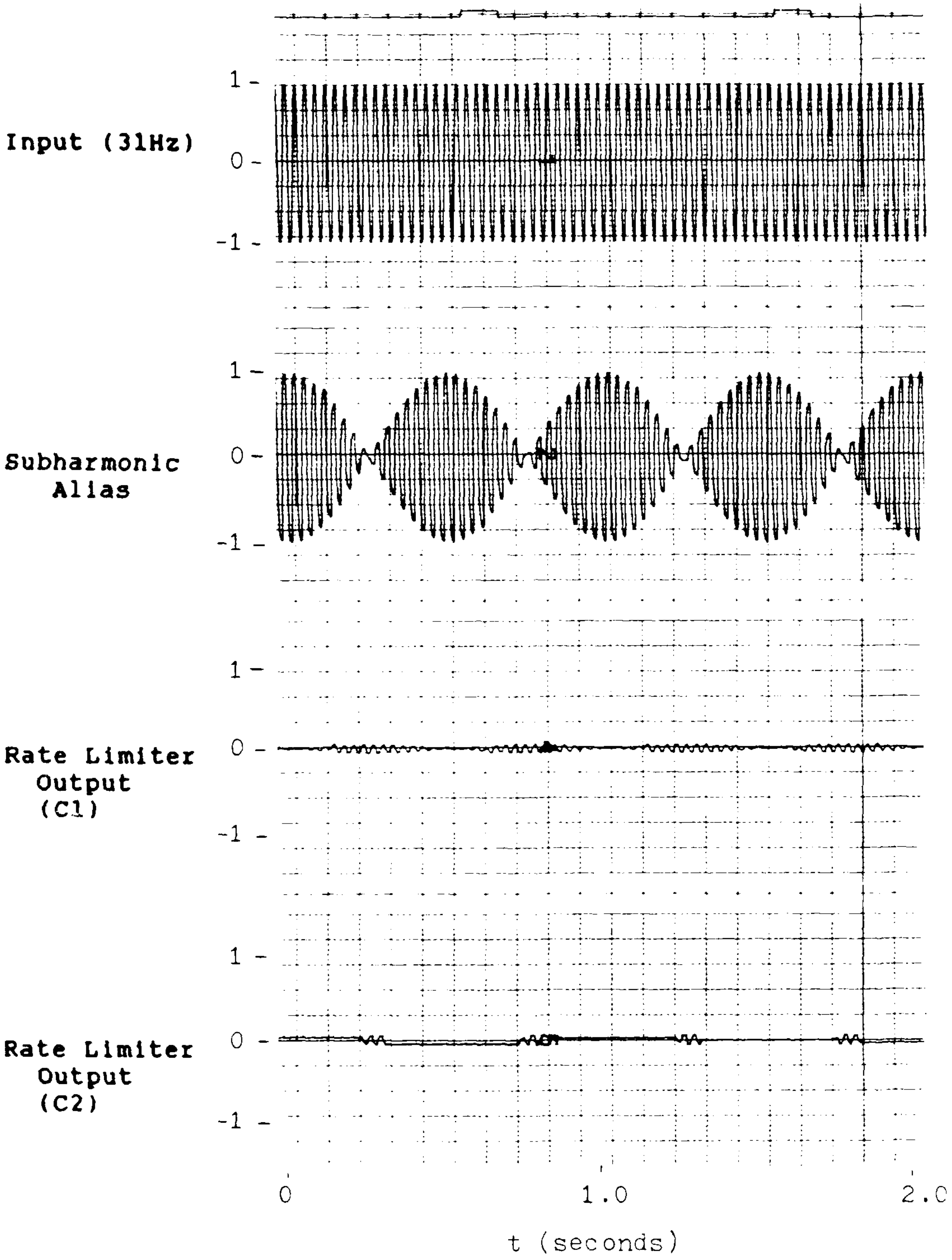

Figure 32. Rate Limiter Output to 31 Hz Input Sampled at 64 samples/sec

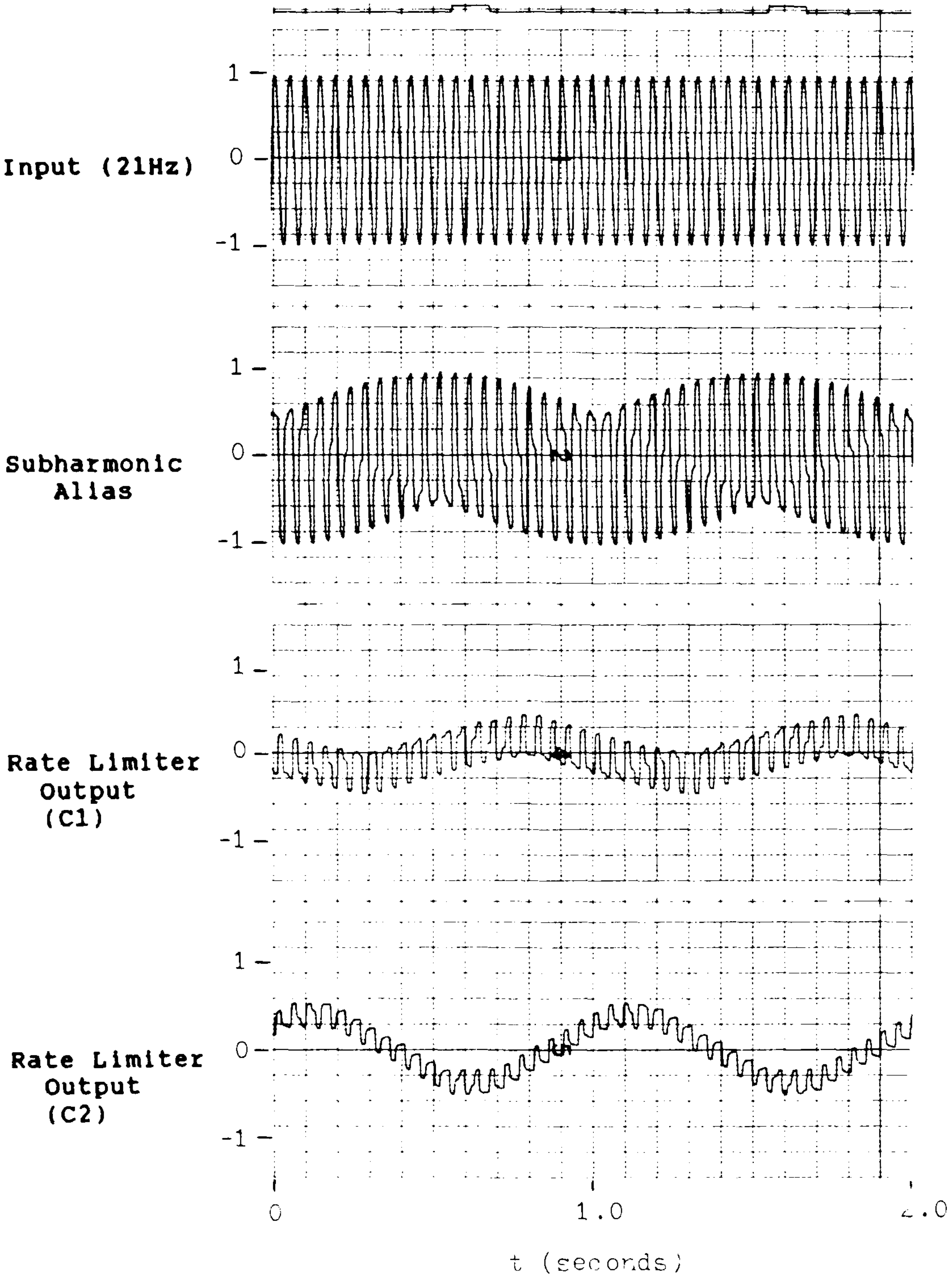

Figure 33. Rate Limiter Output to 21 Hz Input Sampled at 64 samples/second

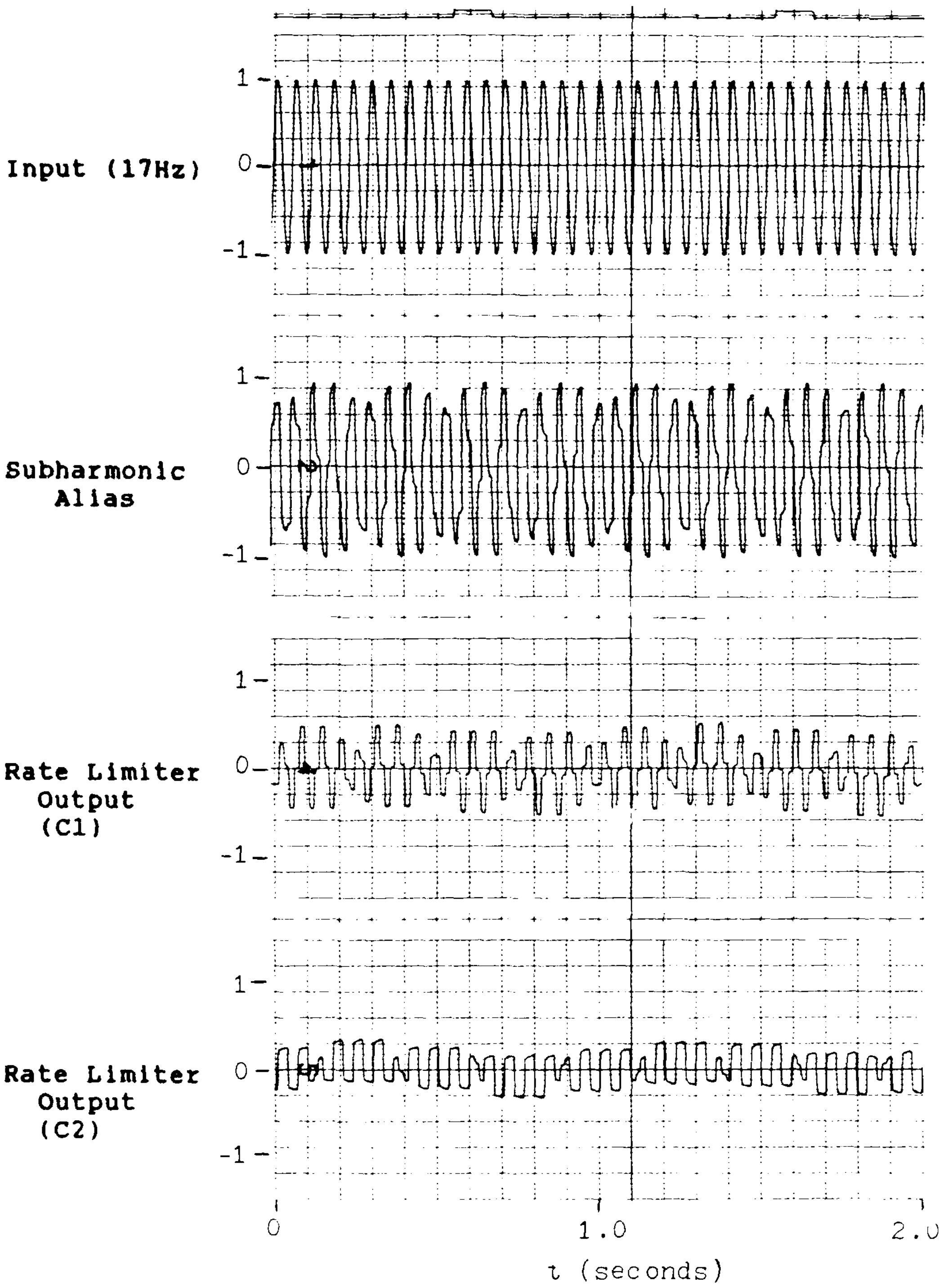

Figure 34. Rate Limiter Output to 17 Hz Input Sampled at 64 samples/second

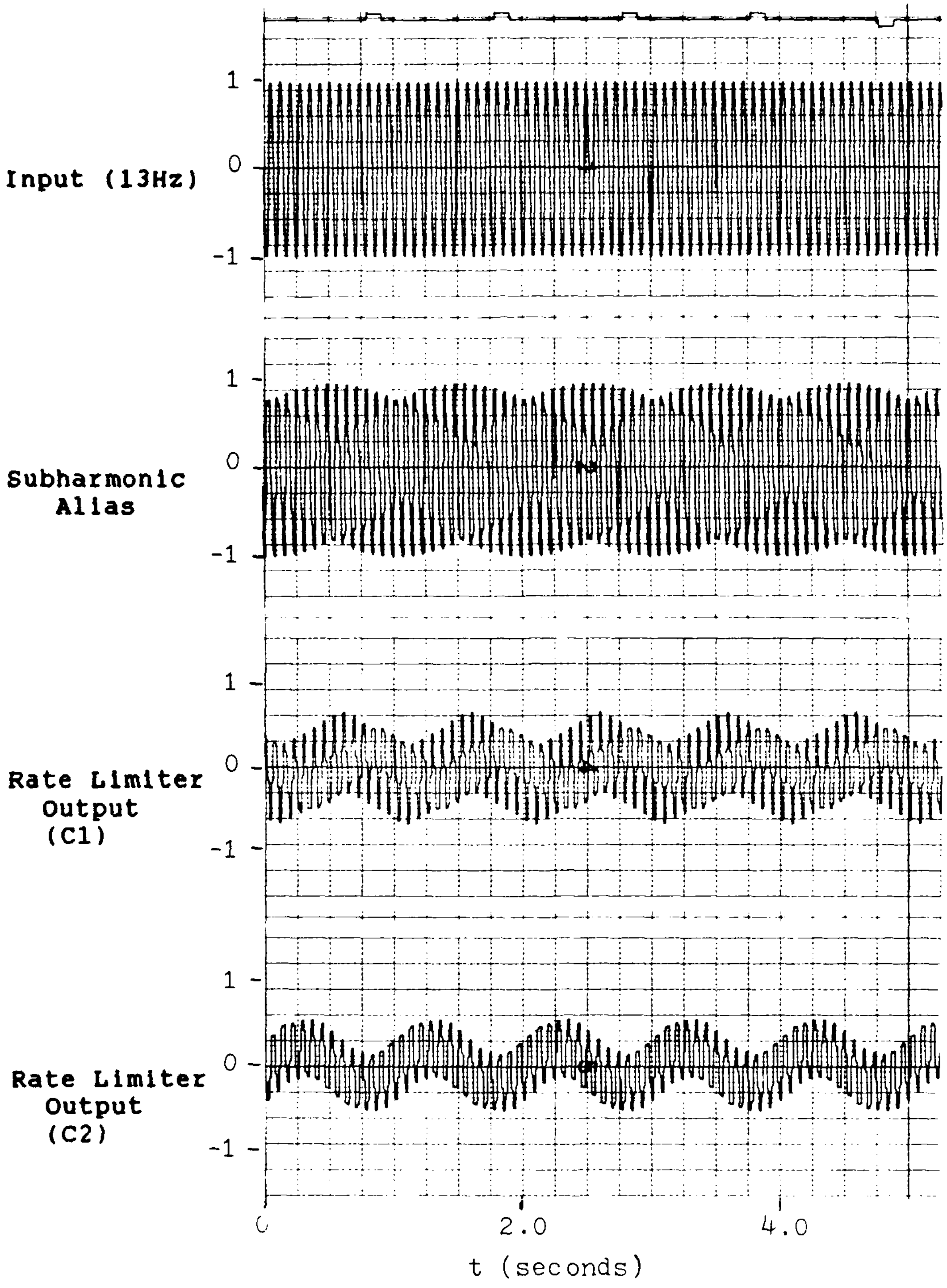

Figure 35. Rate Limiter Output to 13 Hz Input Sampled at 64 samples/second

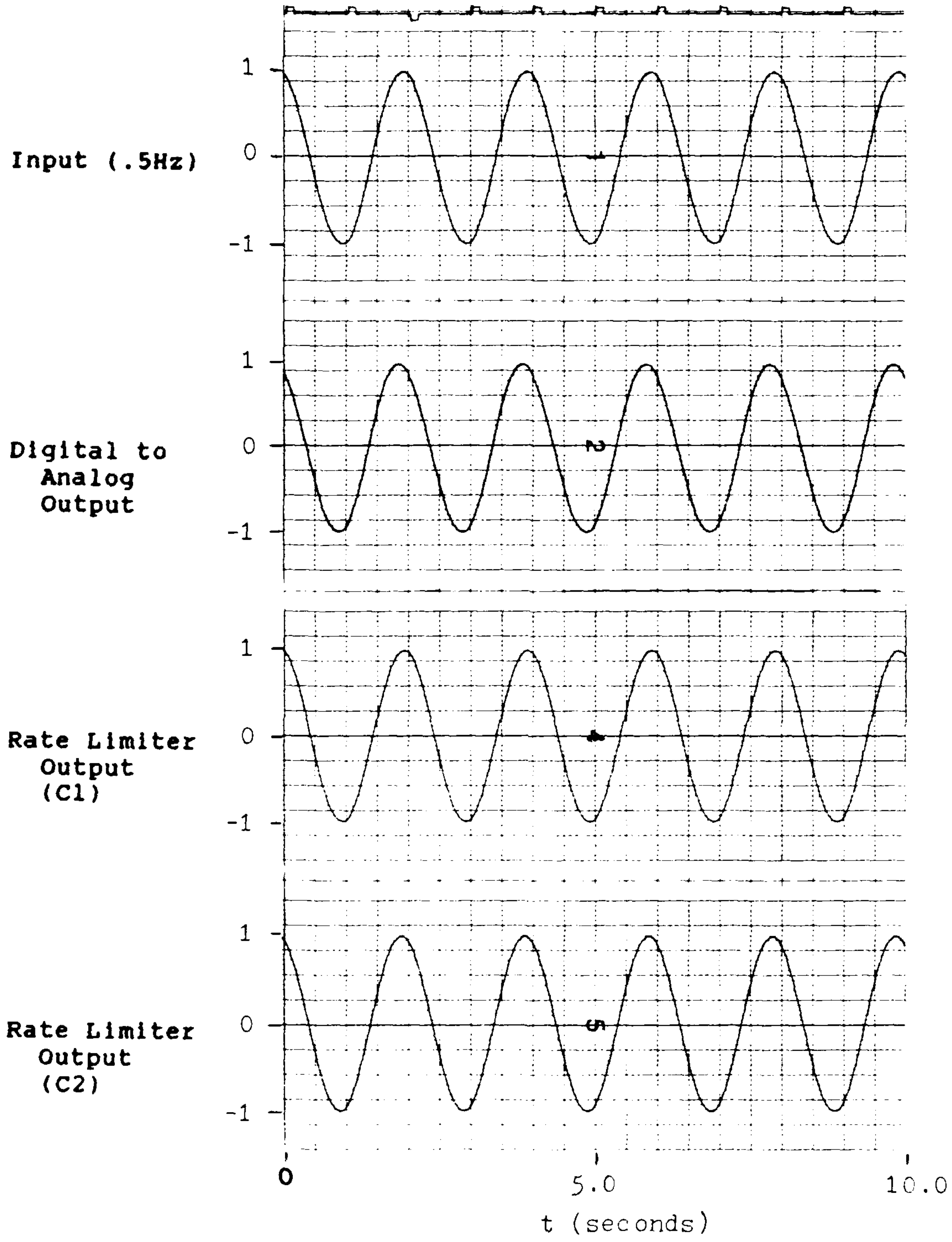

Figure 36. Rate Limiter Output to .5 Hz Input Sampled at 64 samples/second

VII. Conclusions and Recommendations

Conclusions

There are five main points covered in this thesis: first, determine the cause of subharmonic aliasing, as described by the AFTI/F-16 engineers as "the creation of uncorrelated low frequencies whenever a subharmonic of the sample frequency is input into the system"; second, model the subharmonic aliasing effect, so that, by knowing only the input frequency and the system sample rate the output characteristics can be calculated; third, demonstrate the utility of the subharmonic aliasing model when used as an input; fourth, mathematically show how inputs in the subharmonic range ($\omega_s/10 < \omega_o < \omega_s/2$) may induce large interchannel differences; and fifth, demonstrate the effect of the rate limiter on the subharmonic alias and compare two rate limiter configurations.

The mathematical model determined that subharmonic aliasing is the result of impostor frequencies ($\omega_s - \omega_o$, much like aliasing) being introduced into the output signal by the sampling process. The magnitude of the impostor is dependant upon the characteristics of input filters and the

ZOH. Well defined subharmonic aliases occur due to: the impostor frequency and one of two phenomena, either the apparent low frequency surge or the beat frequency. The apparent low frequency surge occurs when the the input frequency is nearly an integer multiple of the sampling frequency. The beat frequency occurs when the input frequency is nearly one half of the sampling frequency.

Simulations from Chapter IV demonstrated the utility of the model as an input used to predict the characteristics of the subharmonic aliased signal.

Mathematical calculations and simulations in Chapter V verify that for even small magnitude inputs large inter-channel differences may occur for frequencies in the subharmonic range.

Simulations discussed in Chapter VI show that it is the combination of well defined subharmonic aliases, the relative high frequencies of the signal, and the nonlinear characteristics of the rate limiter which form the low frequency output from the rate limiter. Simulations also that rate limiting input signals prior to averaging them provid a better overall rate limiting effect.

Recommendations

Since subharmonic aliasing occurs for high frequencies (relative to the sample rate), sampling at a higher rate reduces the undesirable subharmonic aliasing effect. Additionally, since the phenomenon is actually the combination of two high frequencies, using higher order analog filters and/or filters with lower cutoff frequencies will attenuate the input signals in this range.

Unfortunately, the two simple solutions recommended above are not easily implemented, since it is very difficult to change the sample and frame rates once a computer architecture is fixed, and since higher order (and/or lower cutoff frequency) filters tend to reduce the gain and phase margins of the control system.

Further research should involve simulating combinations of the sensor sample rates, Input and Output selector monitor thresholds limits, Rate Limiter configurations, anti-aliasing filter cutoff frequencies, and computation frame rates to determine optimal thresholds for the system.

Appendix A: Examples of Subharmonic Aliasing

Figures A1-A7 are examples of Subharmonic Aliasing with either a beat freqeuncy or an apparent low frequency surge. These plots are generated with an input $r(t)=\cos(2*\pi*f_0*t)$, where f_0 is the input frequency in Hz, to an analog to digital converter and then to a digital to analog converter.

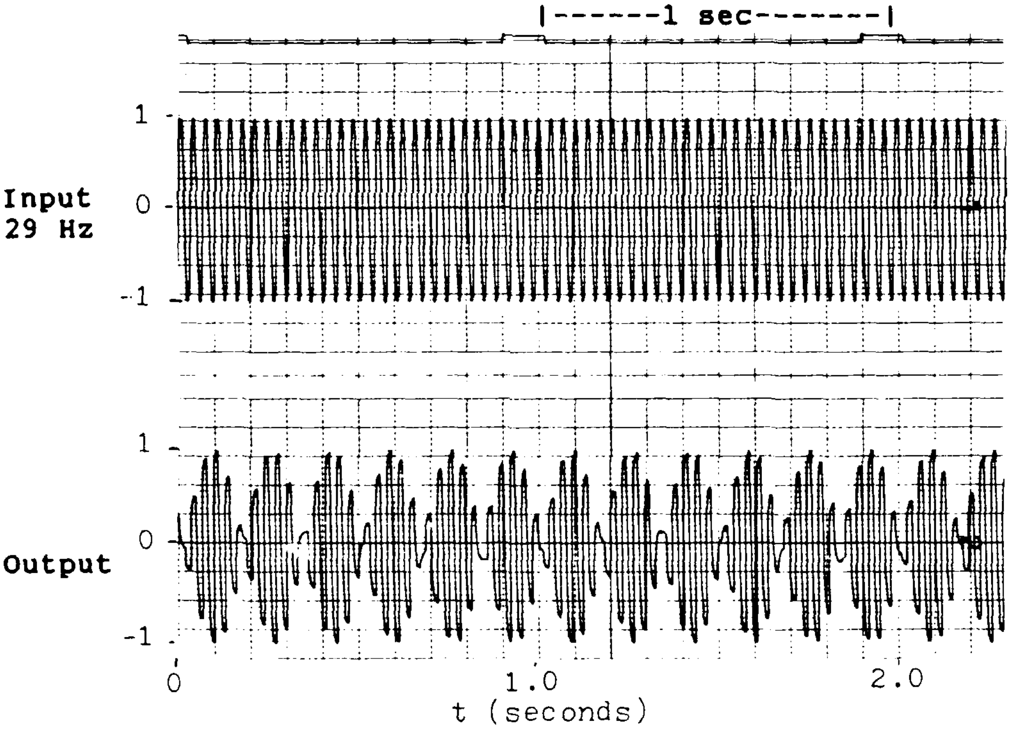

Figure A1. Subharmonic Aliasing Effect (input 29 Hz, sampled at 64 samples/sec)

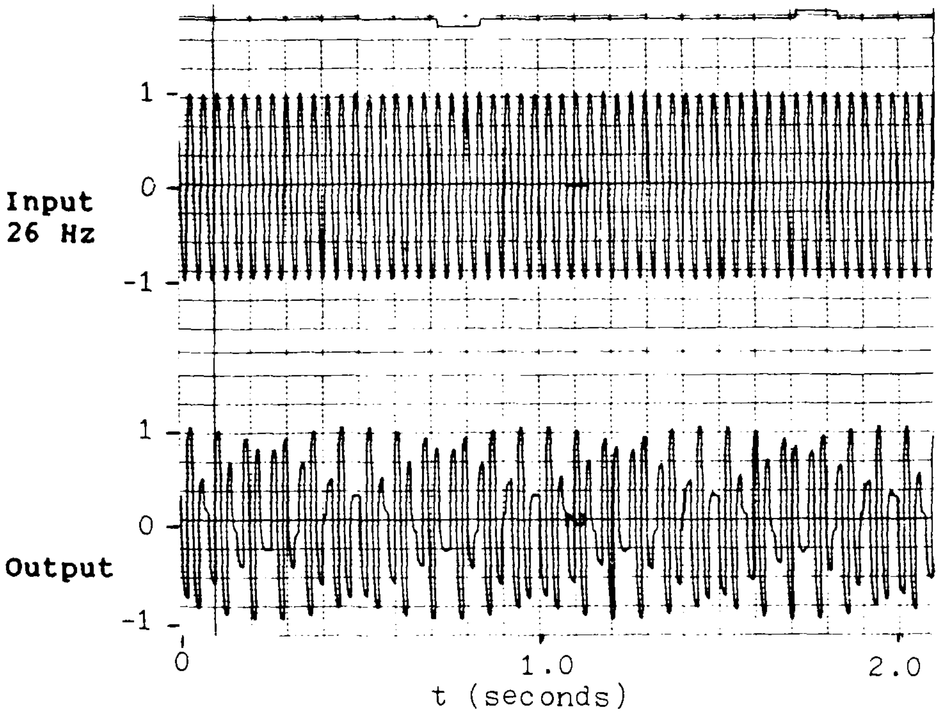

Figure A2. Subharmonic Aliasing Effect (input 26 Hz, sampled at 64 samples/sec)

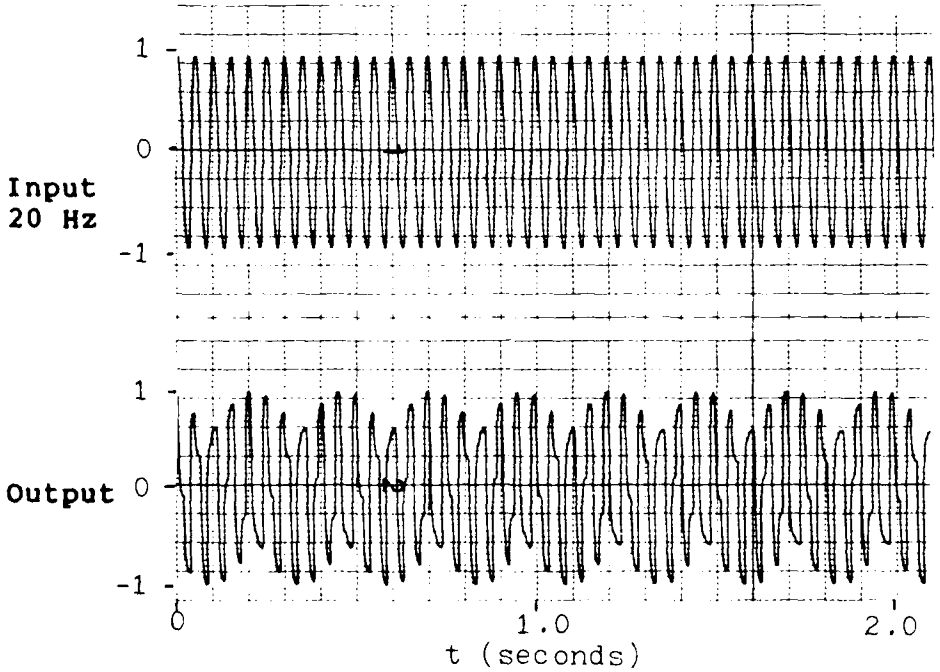

Figure A3. Subharmonic Aliasing Effect (input 20 Hz, sampled at 64 samples/sec)

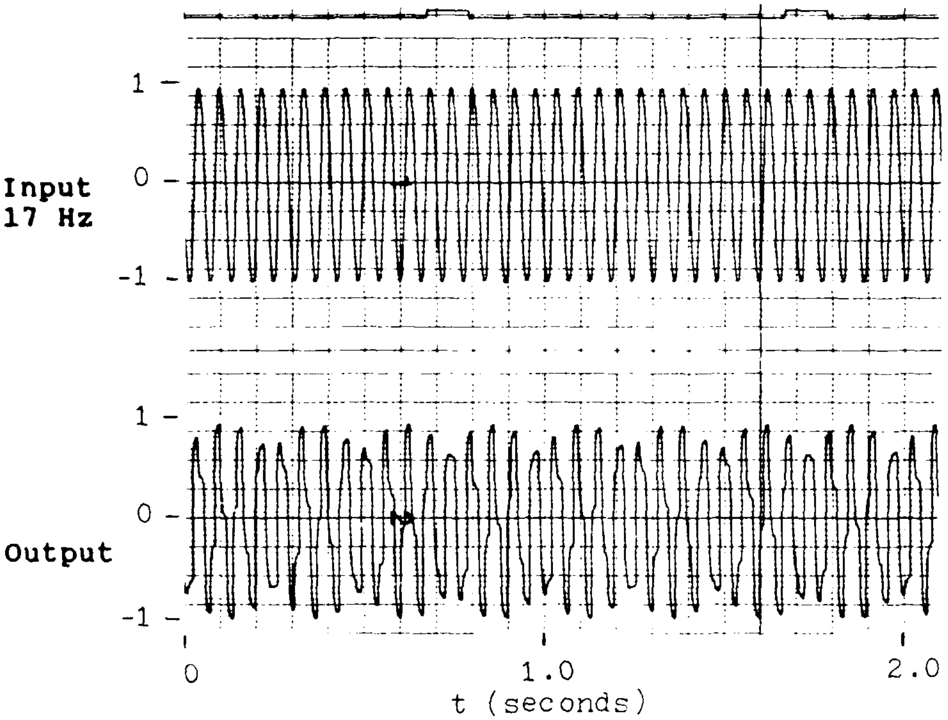

Figure A4. Subharmonic Aliasing Effect (input 17 Hz, sampled at 64 samples/sec)

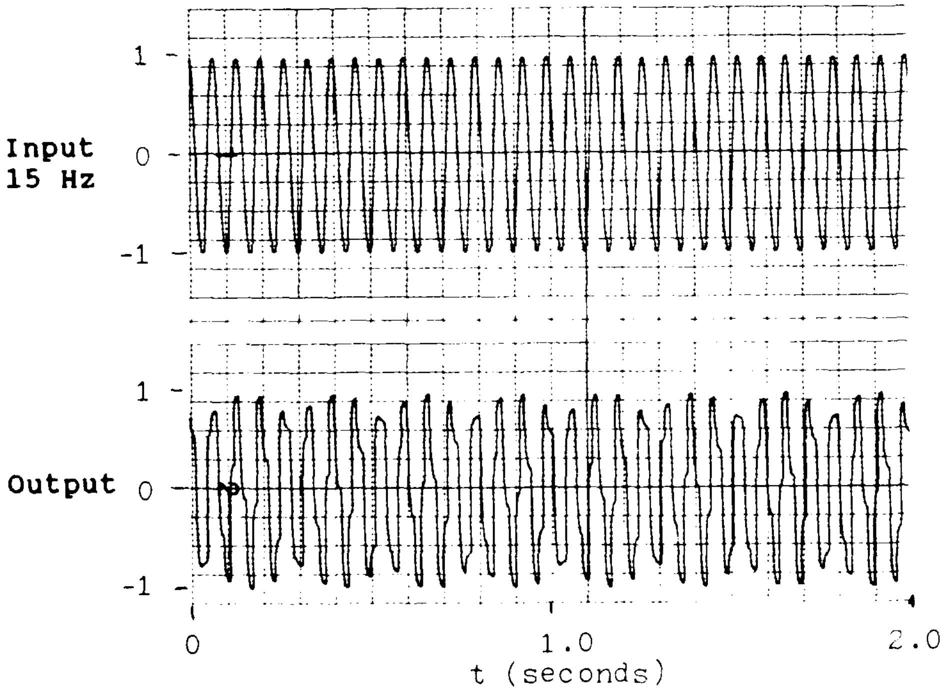

Figure A5. Subharmonic Aliasing Effect (input 15 Hz, sampled at 64 samples/sec)

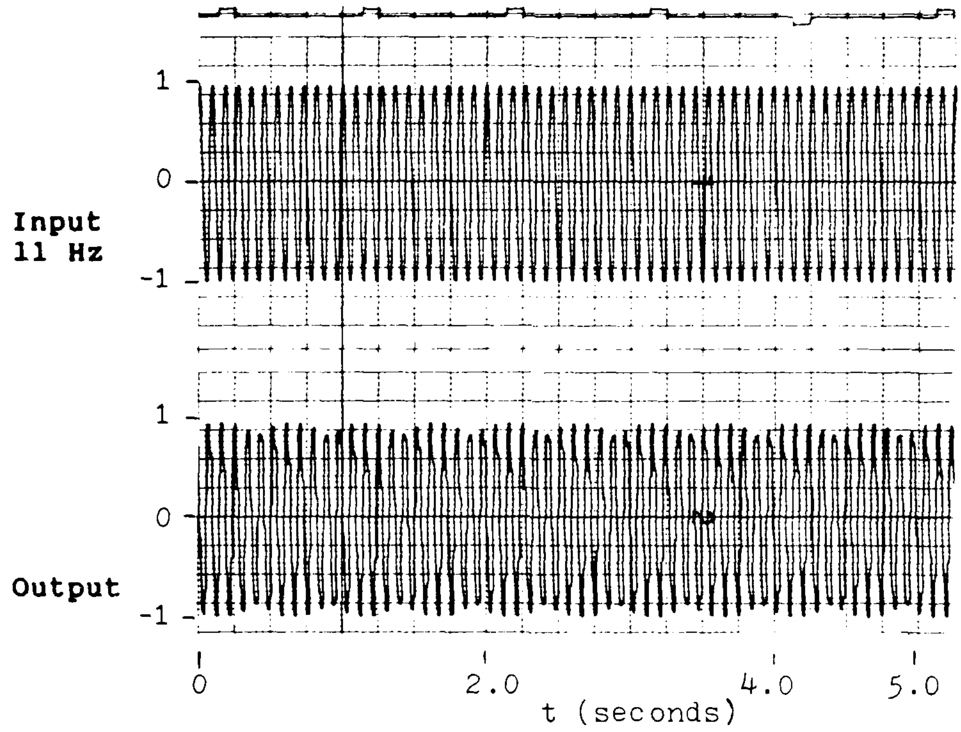

Figure A6. Subharmonic Aliasing Effect (input 11 Hz, sampled at 64 samples/sec)

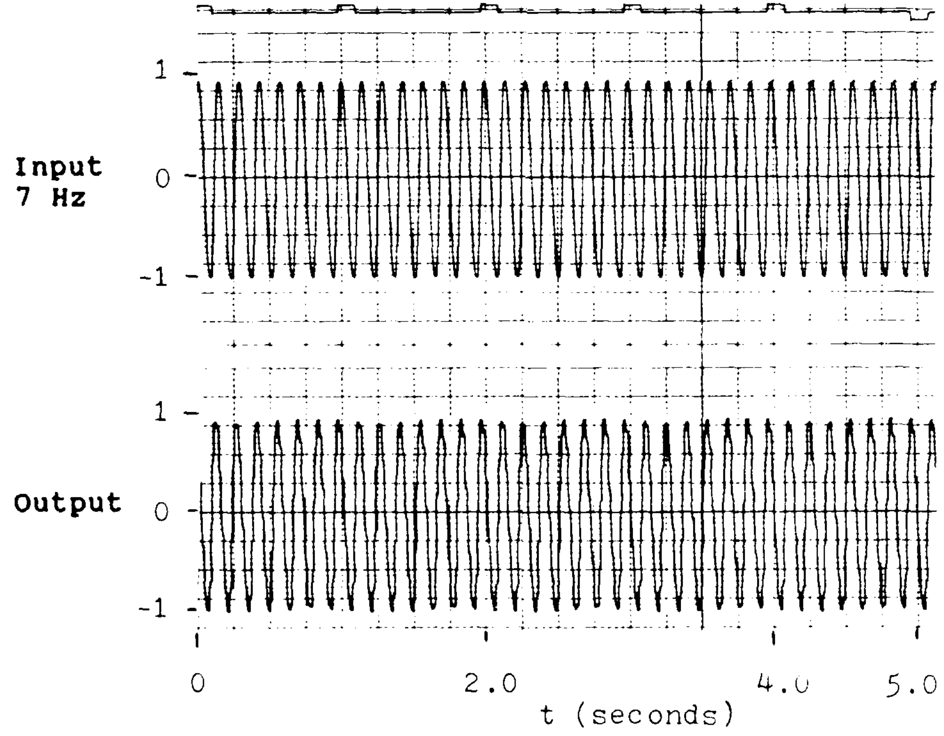

Figure A7. Subharmonic Aliasing Effect (input 7 Hz, sampled at 64 samples/sec)

Figures A8-A13 are examples of a frequency which is a multiple (1/3 in this case) of the sample frequency. These figures demonstrate how the phase if the input determines the magnitude of the output.

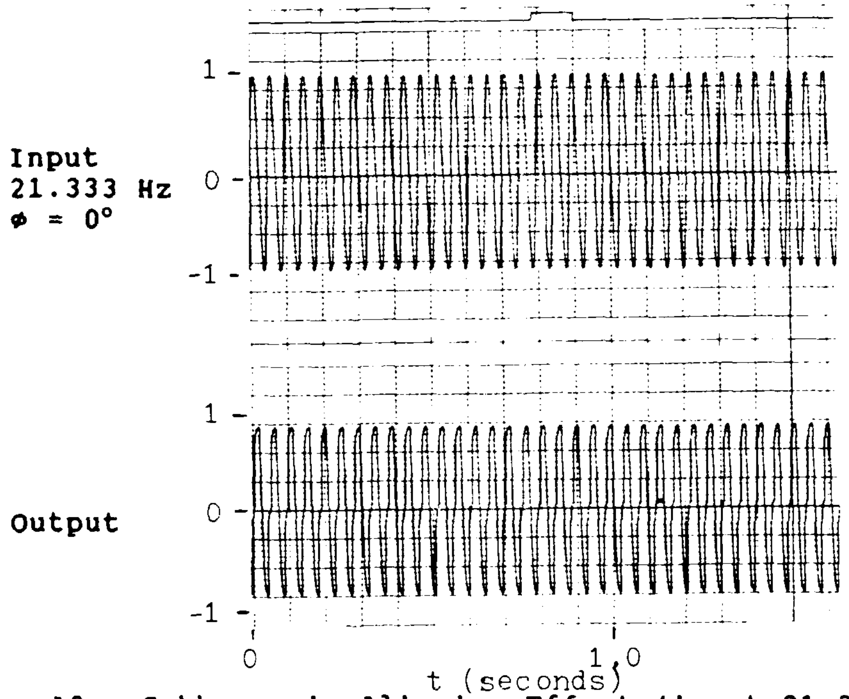

Figure A8. Subharmonic Aliasing Effect (input 21.333 Hz, 0° phase shift, sampled at 64 samples/sec)

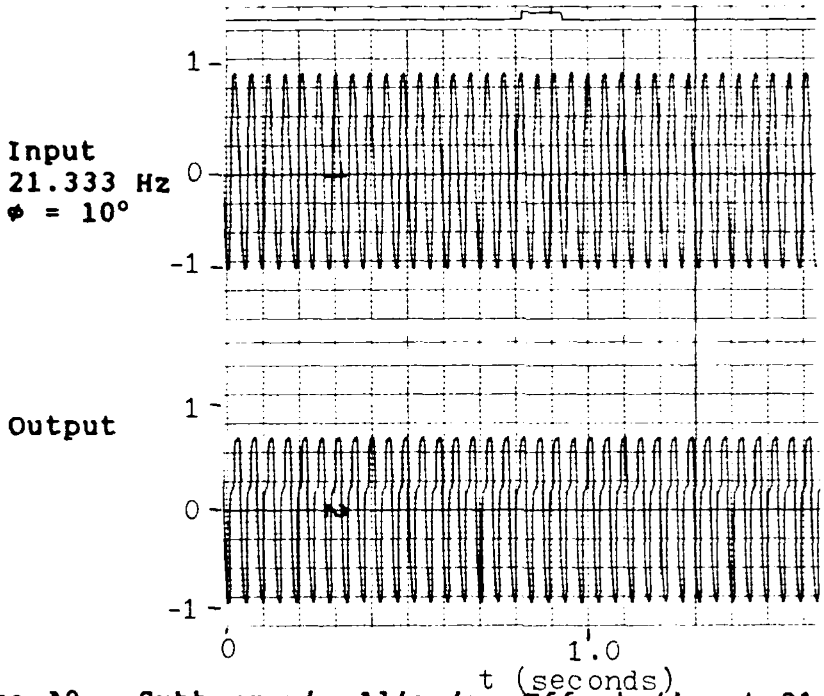

Figure A9. Subharmonic Aliasing Effect (input 21.333 Hz, 10°phase shift sampled at 64 samples/sec)

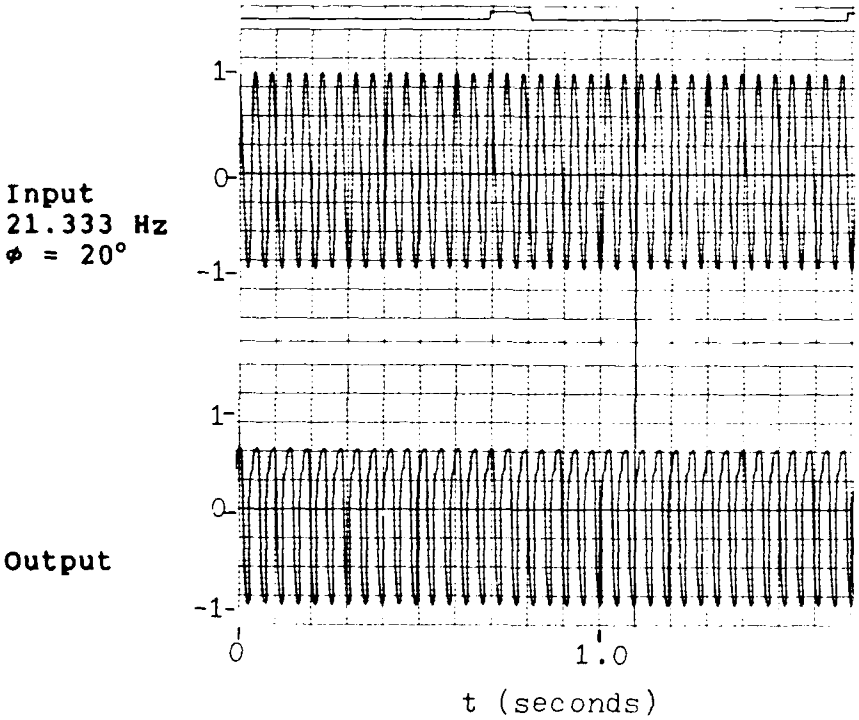

Figure A10. Subharmonic Aliasing Effect (input 21.333 Hz, 20° phase shift, sampled at 64 samples/sec)

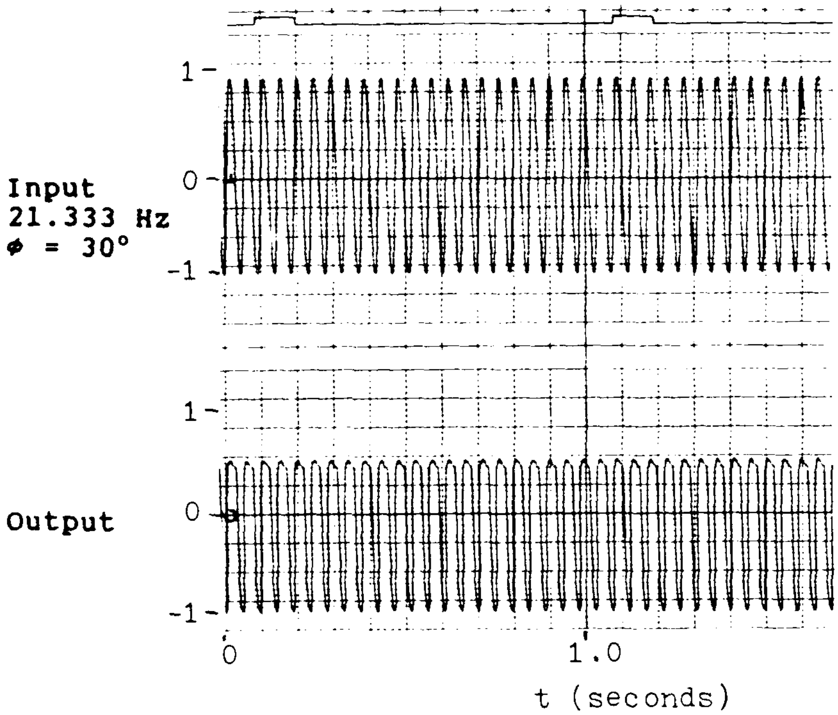

Figure A11. Subharmonic Aliasing Effect (input 21.333 Hz, 30° phased sh'ft, sampled at 64 samples/sec)

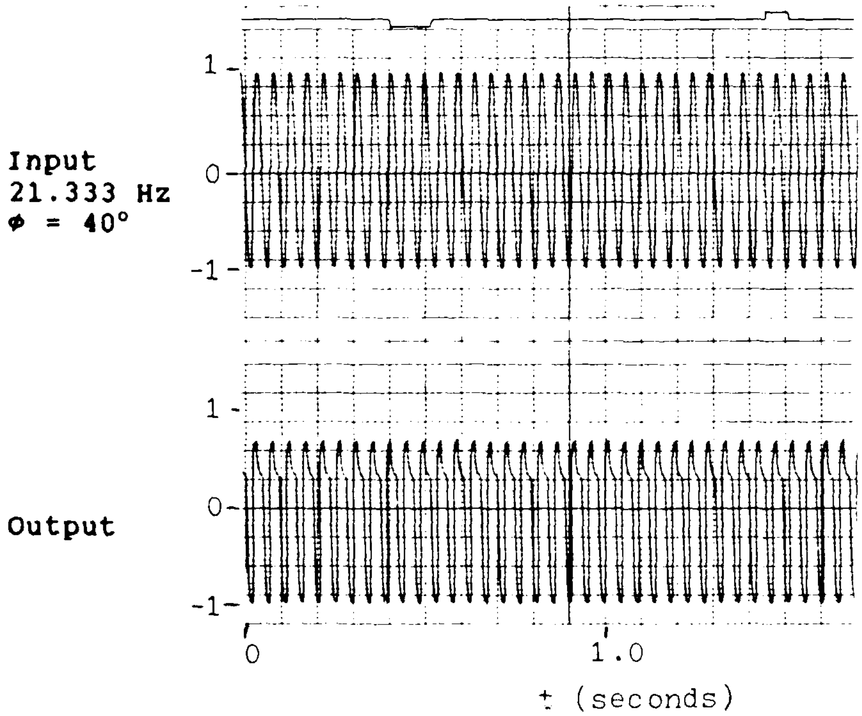

Figure A12. Subharmonic Aliasing Effect (input 21.333 Hz, 40° phased shift, sampled at 64 samples/sec)

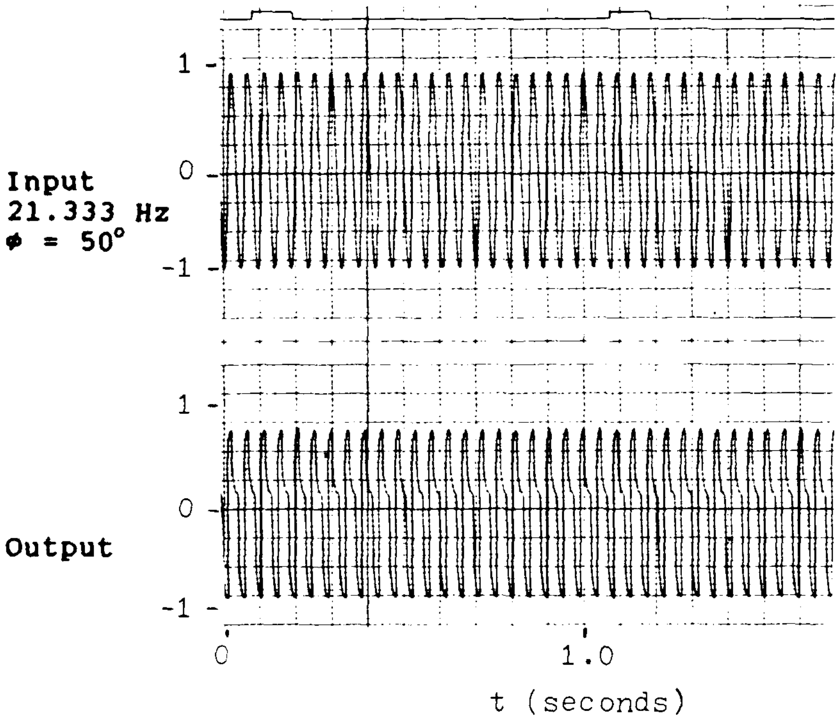

Figure A12. Subharmonic Aliasing Effect (input 21.333 Hz, 50° phased shift, sampled at 64 samples/sec)

Appendix B: SIMSTAR PROGRAMS

The following programs were used to demonstrate subharmonic aliasing, and its effects using the SIMSTSAR hybrid computer. S1.SASIM (Page 88) simulates subharmonic aliasing with a simple A/D - D/A circuit sampling a sinusoidal input. S1.SAFIL (Page 91) simulates the effect of an input filter (FILTER in radians) on the subharmonic alias. S1.TEST5 (Page 93) simulates the effect of an output filter (FILTER in radians) on the subharmonic alias. S1.SATL (Page 95) simulates the maximum interchannel difference due to time skew for user specified threshold levels (TL). S1.RLSIM (Page 97) simulates the effect of a software rate limiter on the subharmonic alias for three specific cases: a single input rate limited, two inputs (the past input and the present input) averaged together and then rate limited, and the inputs rate limited and then averaged together.

```
*PSP=1,0,ERR=ALL
*TITLE
THESIS   SUBHARMONIC ALIASING  SIMULATION
*INPUT
'This is a self contained program and requires no '
'external inputs.  All outputs go to the strip'
'chart recorder'
' '
' '
'If you are typing this program in, do not include
'line numbers'
'THIS PROGRAMS INPUT IS OF R=SIN(2*PI*FREQ*T + PHI)'
'WHERE'
'      FREQ = THE INPUT FREQUENCY IN Hz AND IS SET'
'                 BY THE OPERATOR'
'     PHI = THE PHASE SHIFT IN DEGREES AND IS SET'
'                BY THE OPERATOR'
'   THE SAMPLE RATE IS 64 SAMPLES/SECOND AND MAY BE'
'CHANGED BY THE OPERATOR.'
'  TO DO THIS'
'SET CINT=sample period required'
'SET PERIOD=sample period required'
PROGRAM
  'INTERRUPT DECLARATIONS'
  'INTDEF(0,1,1)'
  'INTDEF(1,1,0)'
INITIAL
    '@BETA(BETA)'
    MAXT = PERIOD/BETA
    LOGPER = CINT * BETA
    FRAD=FREQ*2*PI
    PRAD=PHI*PI/180
    VARIABLE TIME = 0
    CONSTANT BETA=1 , RUNTIM = 3.0 , PERIOD= .015625
    CONSTANT CINT=.015625, FREQ=32, PHI=0,PI=3.14159
    '@PARAMETER BETA,RUNTIM,FREQ,PHI'
    '@MAXVAL FREQ= 64,PHI=360'
    '@MINVAL FREQ=.01,PHI=0'
    '@MAXVAL BETA =100, RUNTIM =50, TIME=50'
    '@MINVAL BETA=.1  , RUNTIM =0 , TIME=0 '
    NSTEPS NSTEP=1
    D1=0
END $'OF INITIAL'
   DYNAMIC
    'INTERRUPT RATE ERROR DECLARATIONS'
         LOGICAL ENDER1, RATER1, ERROR1
         ENDER1 = .FALSE.
         ERROR1 = RATER1
   DERIVATIVE
      CONSTANT D1MAX=1
      CONSTANT A1MAX=1
      CONSTANT CMAX=1
      '@SCALE D1=D1MAX'
      '@SCALE A1=A1MAX'
      '@SCALE C=CMAX'
           D1=A1
   '@PARALLEL'
   R=HARM(0,FRAD,PRAD)
   A1=R
   C=D1
       TERMT(TIME .GT. RUNTIM)
       '@RECORD(REC01,,,,,,,,R,C,A1,D1,,,,)'
    'DEFINE INTERRUPT CONTROL'
          GPI0 = CLOCK(PERIOD)
          GPI1 = CLOCK(LOGPER)
          '@INTRRT 1 = GPI0'
          '@INTRRT 2 = GPI1'
           RATER1=RATERR(GPI0,ENDER1)
   '@END PARALLEL'
   END $'OF DERIVATIVE'
END  $'OF DYNAMIC'
TERMINAL
END $'OF TERMINAL'
END  $'OF PROGRAM'
*TRANSLATE
      DCA(1)=D1
      PADC(1)=A1
*OUTPUT
*END
        SUBROUTINE PREP1
+
        INCLUDE E1.SASIM
        A1=QRPADC(0)*S:A1
        RETURN
        END
C
        SUBROUTINE POST1
+
         INCLUDE E1.SASIM
         COMMON /QQDCP/DCASF(0:0)
         LOGICAL DELAY
         CALL QWDCAR(0,D1*DCASF(0))
         IF (L:RATER1) CALL ZZRTER(1)
         L:ENDER1 =.TRUE.
         DELAY = L:ENDER1
         L:ENDER1 = .FALSE.
         RETURN
         END
C
         SUBROUTINE PREPDCA
+
         COMMON /QQDCP/DCASF(0:0)
         DCASF(0) = 1.0/QDCASR(0)/D1MAX
         RETURN
         END
C
```

```
*PSP=1,0,ERR=ALL
*TITLE
EFFECTS OF FIRST ORDER ANTI-ALIASING FILTER ON SA
*INPUT
PROGRAM
  'INTERRUPT DECLARATIONS'
  'INTDEF(0,1,1)'
  'INTDEF(1,1,0)'
INITIAL
    '@BETA(BETA)'
    MAXT = PERIOD/BETA
    LOGPER = CINT * BETA
    FRAD=FREQ*2*PI
    POLE=1/FILTER
    VARIABLE TIME = 0
    CONSTANT BETA=1 , RUNTIM = 3.0 , PERIOD= .015625
    CONSTANT CINT=.015625, FREQ=32, PI=3.14159
    CONSTANT FILTER=38.4,K=1
    '@PARAMETER BETA,RUNTIM,FREQ,FILTER,K'
    '@MAXVAL FREQ= 64, FILTER=200,K=1000'
    '@MINVAL FREQ=.01, FILTER=.01,K=1'
    '@MAXVAL BETA =100, RUNTIM =50, TIME=50'
    '@MINVAL BETA=.1  , RUNTIM =0 , TIME=0 '
    NSTEPS NSTEP=1
    D1=0
END $'OF INITIAL'
   DYNAMIC
    'INTERRUPT RATE ERROR DECLARATIONS'
         LOGICAL ENDER1, RATER1, ERROR1
         ENDER1 = .FALSE.
         ERROR1 = RATER1
   DERIVATIVE
      CONSTANT D1MAX=1
      CONSTANT A1MAX=1
      CONSTANT CMAX=1
      '@SCALE D1=D1MAX'
      '@SCALE A1=A1MAX'
      '@SCALE C=CMAX'
          D1=A1
   '@PARALLEL'
   R=HARM(0,FRAD,0)
   A1=REALPL(POLE,R,0)
   C=D1
       TERMT(TIME .GT. RUNTIM)
       '@RECORD(REC01,,,,,,,,R,C,A1,D1,,,,)'
    'DEFINE INTERRUPT CONTROL'
          GPI0 = CLOCK(PERIOD)
          GPI1 = CLOCK(LOGPER)
          '@INTRRT 1 = GPI0'
          '@INTRRT 2 = GPI1'
           RATER1=RATERR(GPI0,ENDER1)
   '@END PARALLEL'
   END $'OF DERIVATIVE'
END  $'OF DYNAMIC'
TERMINAL
END $'OF TERMINAL'
END  $'OF PROGRAM'
*TRANSLATE
      DCA(1)=D1
      PADC(1)=A1
*OUTPUT
*END
        SUBROUTINE PREP1
+
        INCLUDE E1.SAFIL
        A1=QRPADC(0)*S:A1
        RETURN
        END
C
        SUBROUTINE POST1
+
         INCLUDE E1.SAFIL
         COMMON /QQDCP/DCASF(0:0)
         LOGICAL DELAY
         CALL QWDCAR(0,D1*DCASF(0))
         IF (L:RATER1) CALL ZZRTER(1)
         L:ENDER1 =.TRUE.
         DELAY = L:ENDER1
         L:ENDER1 = .FALSE.
         RETURN
         END
C
         SUBROUTINE PREPDCA
+
         COMMON /QQDCP/DCASF(0:0)
         DCASF(0) = 1.0/QDCASR(0)/D1MAX
         RETURN
         END
```

```
*PSP=1,0,ERR=ALL
*TITLE
HARDWARE SIMULATION WITH NOISE AND OUTPUT FILTER
*INPUT
PROGRAM
  'INTERRUPT DECLARATIONS'
  'INTDEF(0,1,1)'
  'INTDEF(1,1,0)'
INITIAL
    '@BETA(BETA)'
    MAXT = PERIOD/BETA
    LOGPER = CINT * BETA
    FRAD=FREQ*2*PI
    POLE=1/FILTER
    VARIABLE TIME = 0
    '@EXTERN NOISE'
    CONSTANT BETA=1 , RUNTIM = 3.0 , PERIOD= .015625
    CONSTANT CINT=.015625, FREQ=32,   PI=3.14159
    CONSTANT FILTER=38.4,K=1, MNOISE=1.0
    '@PARAMETER BETA,RUNTIM,FREQ,FILTER,K'
    '@MAXVAL FREQ= 64, FILTER=200,K=100'
    '@MINVAL FREQ=.01, FILTER=.01,K=1'
    '@MAXVAL BETA =100, RUNTIM =50, TIME=50'
    '@SCALE NOISE=MNOISE'
    '@MINVAL BETA=.1  , RUNTIM =0 , TIME=0 '
    NSTEPS NSTEP=1
    D1=0
END $'OF INITIAL'
   DYNAMIC
    'INTERRUPT RATE ERROR DECLARATIONS'
         LOGICAL ENDER1, RATER1, ERROR1
         ENDER1 = .FALSE.
         ERROR1 = RATER1
   DERIVATIVE
      CONSTANT D1MAX=1.2
      CONSTANT FMAX=1.2
      CONSTANT CMAX=1.2
      '@SCALE D1=D1MAX'
      '@SCALE FILOUT=FMAX'
      '@SCALE C=CMAX'
      PROCEDURAL(D1=A1)
           D1=A1
      END
   '@PARALLEL'
   R=HARM(0,FRAD,0)
   RN = R + NOISE
   A1=RN
   FILOUT=REALPL(POLE,D1,0)
   C=FILOUT*K
       TERMT(TIME .GT. RUNTIM)
       '@RECORD(REC01,,,,,,,,R,C,A1,D1,NOISE,,,)'
    'DEFINE INTERRUPT CONTROL'
          GPIO = CLOCK(PERIOD)
          GPI1 = CLOCK(LOGPER)
          '@INTRRT 1 = GPI0'
          '@INTRRT 2 = GPI1'
           RATER1=RATERR(GPI0,ENDER1)
   '@END PARALLEL'
   END $'OF DERIVATIVE'
END  $'OF DYNAMIC'
TERMINAL
END $'OF TERMINAL'
END  $'OF PROGRAM'
*TRANSLATE
      DCA(1)=D1
      PADC(1)=A1
      CONNECT AIN31=NOISE
*OUTPUT
*END
        SUBROUTINE PREP1
+
        INCLUDE E1.TEST5
        A1=QRPADC(0)*S:A1
        RETURN
        END
C
        SUBROUTINE POST1
+
         INCLUDE E1.TEST5
         COMMON /QQDCP/DCASF(0:0)
         LOGICAL DELAY
         CALL QWDCAR(0,D1*DCASF(0))
         IF (L:RATER1) CALL ZZRTER(1)
         L:ENDER1 =.TRUE.
         DELAY = L:ENDER1
         L:ENDER1 = .FALSE.
         RETURN
         END
C
         SUBROUTINE PREPDCA
+
         COMMON /QQDCP/DCASF(0:0)
         DCASF(0) = 1.0/QDCASR(0)/D1MAX
         RETURN
         END
```

```
*PSP=1,0,ERR=ALL
*TITLE
THESIS S1.SATL-INPUT SELECTOR/MONITOR
*INPUT
PROGRAM
'THIS PROGRAM SIMULATES A DUAL REDUNDANT,'
' ASYNCHRONOUS SYSTEM BY FINDING THE DIFFERENCE OF'
' THE INPUT AND THE INPUT DELAYED ONE SAMPLE.  '
'SINCE THE AFTI/F-16 RUNS AT 256 SAMPLES/SECOND,'
' THE SAMPLE TIME IS .00390625 SEC. THIS SIMULATES'
' THE GREATEST TIME SKEW BETWEEN COMPUTERS'
'OF 0.00390625 SECONDS'
  'INTERRUPT DECLARATIONS'
  'INTDEF(0,1,1)'
  'INTDEF(1,1,0)'
INITIAL
    '@BETA(BETA)'
    MAXT = PERIOD/BETA
    LOGPER = CINT * BETA
    FRAD=FREQ*2*PI
    VARIABLE TIME = 0
    CONSTANT BETA=1,RUNTIM = 3.0,PERIOD= .00390625
    CONSTANT CINT=.00390625, FRAD=131.9,FREQ=21
    CONSTANT M=3, PI=3.14159
    '@PARAMETER BETA,RUNTIM,FREQ'
    '@MAXVAL FREQ=256'
    '@MINVAL FREQ=.01'
    '@MAXVAL BETA =100, RUNTIM =50, TIME=50'
    '@MINVAL BETA=.1  , RUNTIM =0 , TIME=0 '
    NSTEPS NSTEP=1
    D1=0
    D2=0
    DIF =0
END $'OF INITIAL'
   DYNAMIC
    'INTERRUPT RATE ERROR DECLARATIONS'
         LOGICAL ENDER1, RATER1, ERROR1
         ENDER1 = .FALSE.
         ERROR1 = RATER1
   DERIVATIVE
      CONSTANT D1MAX=3.1,D2MAX=3.1,DIFMAX=3.1
      CONSTANT A1MAX=3.1
      '@SCALE D1=D1MAX'
      '@SCALE D2=D2MAX'
      '@SCALE DIF=DIFMAX'
      '@SCALE A1=A1MAX'
      PROCEDURAL(D1,D2,DIF =A1)
         '@IMPL(D1)'
          DIF=D2-D1
          D2=D1
          D1=A1
        '@END IMPL'
      END
   '@PARALLEL'
   R=HARM(0,FRAD,0)
   A1=M*R
   C=D1
       TERMT(TIME .GT. RUNTIM)
       '@RECORD(REC01,,,,,,,,R,DIF,D2,,,,,)'
    'DEFINE INTERRUPT CONTROL'
          GPI0 = CLOCK(PERIOD)
          GPI1 = CLOCK(LOGPER)
          '@INTRRT 1 = GPI0'
          '@INTRRT 2 = GPI1'
           RATER1=RATERR(GPI0,ENDER1)
   '@END PARALLEL'
   END $'OF DERIVATIVE'
END  $'OF DYNAMIC'
TERMINAL
END $'OF TERMINAL'
END  $'OF PROGRAM'
*TRANSLATE
      DCA(1)=D1
      PADC(1)=A1
*OUTPUT
*END
        SUBROUTINE PREP1
+
        INCLUDE E1.SATL
        A1=QRPADC(0)*S:A1
        RETURN
        END
C
        SUBROUTINE POST1
+
         INCLUDE E1.SATL
         COMMON /QQDCP/DCASF(0:2)
         LOGICAL DELAY
         CALL QWDCAR(0,D1*DCASF(0))
         CALL QWDCAR(1,D2*DCASF(1))
         CALL QWDCAR(2,DIF *DCASF(2))
         IF (L:RATER1) CALL ZZRTER(1)
         L:ENDER1 =.TRUE.
         DELAY = L:ENDER1
         L:ENDER1 = .FALSE.
         RETURN
         END
C
         SUBROUTINE PREPDCA
+
         COMMON /QQDCP/DCASF(0:0)
         DCASF(0) = 1.0/QDCASR(0)/D1MAX
         DCASF(1) = 1.0/QDCASR(1)/D2MAX
         DCASF(2) = 1.0/QDCASR(2)/DIFMAX
         RETURN
         END
```

```
*PSP=1,0,ERR=ALL
*TITLE
THESIS  S1.RLSIM   RATE LIMITER
*INPUT
PROGRAM
'THIS PROGRAM SIMULATES 1: THE  RATE LIMITER'
' AFTER THE OUTPUT S/M WITH NO INTER-CHANNEL '
'DIFFERENCE BETWEEN COMPUTERS (RATE1),'
'2: THE RATE LIMITER AFTER THE OUTPUT S/M WITH '
'MAXIMUM INTER-CHANNEL DIFFERENCE (RATE2)'
'AND 3: RATE LIMITERS PRIOR TO THE OUTPUT S/M WITH'
'MAXIMUM INTER-CHANNEL DIFFERENCE (RATE3)'
'THE SAMPLE PERIOD IS .015625 SEC, WHICH SIMULATES'
'THE AFTI COMPUTATIONAL FRAME RATE'
'  '
  'INTERRUPT DECLARATIONS'
  'INTDEF(0,1,1)'
  'INTDEF(1,1,0)'
INITIAL
    '@BETA(BETA)'
    MAXT = PERIOD/BETA
    LOGPER = CINT * BETA
    FRAD=FREQ*2*PI
    VARIABLE TIME = 0
    CONSTANT BETA=1, RUNTIM=3.0, PERIOD=.015625
    CONSTANT TL=.45, PI=3.14159
    CONSTANT CINT=.015625, FRAD = 131.9, FREQ=21
    '@PARAMETER BETA,RUNTIM,FREQ,TL'
    '@MAXVAL FREQ=256, TL=1'
    '@MINVAL FREQ=.01, TL=0'
    '@MAXVAL BETA =100, RUNTIM =50, TIME=50'
    '@MINVAL BETA=.1  , RUNTIM =0 , TIME=0 '
    NSTEPS NSTEP=1
    D1=0
     N=0
END $'OF INITIAL'
   DYNAMIC
    'INTERRUPT RATE ERROR DECLARATIONS'
         LOGICAL ENDER1, RATER1, ERROR1
         ENDER1 = .FALSE.
         ERROR1 = RATER1
   DERIVATIVE
      CONSTANT D1MAX=1, R1MAX=1, R2MAX=1, R3MAX=1
      CONSTANT A1MAX=1.0
      '@SCALE D1=D1MAX'
      '@SCALE RATE1=R1MAX'
      '@SCALE RATE2=R2MAX'
      '@SCALE RATE3=R3MAX'
      '@SCALE A1=A1MAX'
      PROCEDURAL(RL1,RL2,RL3 = D1,TL,N)
          D1=A1
          CALL RL(RL1,RL2,RL3,D1,TL,N)
          RATE1=RL1
          RATE2=RL2
          RATE3=RL3
        END
   '@PARALLEL'
   R=HARM(0,FRAD,0)
   A1=R
   C=D1
    TERMT(TIME .GT. RUNTIM)
    '@RECORD(REC01,,,,,,,,R,C,RATE1,RATE2,RATE3,,,)'
    'DEFINE INTERRUPT CONTROL'
          GPI0 = CLOCK(PERIOD)
          GPI1 = CLOCK(LOGPER)
          '@INTRRT 1 = GPI0'
          '@INTRRT 2 = GPI1'
           RATER1=RATERR(GPI0,ENDER1)
   '@END PARALLEL'
   END $'OF DERIVATIVE'
END  $'OF DYNAMIC'
TERMINAL
END $'OF TERMINAL'
END  $'OF PROGRAM'
*TRANSLATE
      DCA(1)=D1
      PADC(1)=A1
*OUTPUT
*END
        SUBROUTINE PREP1
+
        INCLUDE E1.RLSIM
        A1=QRPADC(0)*S:A1
        RETURN
        END
C
        SUBROUTINE POST1
+
         INCLUDE E1.RLSIM
         COMMON /QQDCP/DCASF(0:2)
         LOGICAL DELAY
         CALL QWDCAR(0,D1*DCASF(0))
         CALL QWDCAR(1,RATE1*DCASF(1))
         CALL QWDCAR(2,RATE2 *DCASF(2))
         CALL QWDCAR(3,RATE3 *DCASF(3))
         IF (L:RATER1) CALL ZZRTER(1)
         L:ENDER1 =.TRUE.
         DELAY = L:ENDER1
         L:ENDER1 = .FALSE.
         RETURN
         END
C
         SUBROUTINE PREPDCA
+
         COMMON /QQDCP/DCASF(0:0)
         DCASF(0) = 1.0/QDCASR(0)/D1MAX
         DCASF(1) = 1.0/QDCASR(1)/R1MAX
         DCASF(2) = 1.0/QDCASR(2)/R2MAX
         DCASF(3) = 1.0/QDCASR(3)/R3MAX
         RETURN
         END
C
         SUBROUTINE RL(R1,R2,R3,Z1,TLEV,C)
*
         IF (C .EQ. 0) THEN
           R=0
           R1=0
           R2=0
           Z2=Z1
         END IF
         C=C+1
         DIF1=R1-Z1
           IF (DIF1 .LE. -TLEV) THEN
              R1=R1+TLEV
           ELSE IF (DIF1 .GE. TLEV) THEN
              R1=R1-TLEV
           ELSE
              R1=Z1
           END IF
C
         AVE2=(Z2+Z1)/2
          DIF2=R2-AVE2
           IF (DIF2 .LE. -TLEV) THEN
              R2=R2+TLEV
           ELSE IF (DIF2 .GE. TLEV) THEN
              R2=R2-TLEV
           ELSE
              R2=AVE2
           END IF
         DIF = R-Z2
           IF (DIF .LE. -TLEV) THEN
              R=R+TLEV
           ELSE IF (DIF .GE. TLEV) THEN
              R=R-TLEV
           ELSE
              R=Z2
           END IF
         R3=(R1+R)/2
         Z2=Z1
         RETURN
         END
```

Appendix C: Resources Required

The following is a list of resources used to simulate and study subharmonic aliasing.

a) Cyber Computer (Total CAD Package)

b) ICECAP CAD Package

c) SIMSTAR Hybrid Computer

d) Strip Chart Recorder

e) MATRIX

Bibliography

1. Rhodes, 1Lt Michael L. and Terrence M. Shindler. Design Considerations for Asynchronous Digital Flight Control Systems. Research Paper. AFTI/F-16 System Program Office, Wright-Patterson AFB, OH, May 1982 (ASD 82 1073).

2. Serpen, 1Lt Gursel, TUAF. Failure Detection and Isolation For an Asynchronous Digital Flight Control System, MS Thesis. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH. (March 1987).

3. AFWAL. AFTI/F-16 Development and Integration Program DFCS Phase Final Technical Report. AFWAL-TR-84-3008; Volume 2. Air Force Systems Command: Wright-Patterson Air Force Base, Ohio, Dec 1984.

4. Arabian, Alan M. and others. "AFTI/F-16 Digital Flight Control Computer Design," Proceedings of the IEEE 1983 NAECON. 1426-1432. New York: IEEE Press, 1983.

5. Yousey, W. J. and others. "AFTI/F-16 DFCS Development Summary - A Report to Industry, Redundancy Management System Design," Proceedings of the IEEE 1893 NAECON. 1220-1226. New York: IEEE Press, 1983

6. AFWAL. AFTI/F-16 Development and Integration Program DFCS Phase Final Technical Report. AFWAL-TR-84-3008; Volume 3, Part 3. Air Force Systems Command: Wright-Patterson Air Force Base, Ohio, Dec 1984.

7. Schindler, T. M. and others. "AFTI/F-16 DFCS Development Summary - A Report to Industry, Software Design/Mechanization," Proceedings of the IEEE 1983 NAECON. 1227-1234. New York: IEEE Press, 1983

8. Houpis, Constantine H. and Gary B. Lamont. Digital Control Systems. Mcgraw-Hill Book Company, New York, 236-237. 1985.

9. Oppenheim, Alan V. and Alan S. Willsky. Signals and Systems. Prentice-Hall Inc, Englewood Cliffs, New Jersey. 1983.

10. Kuo, B. C. Discrete-Data Control Systems. Prentice-Hall Inc, Englewood Cliffs, New Jersey. 11-12. 1970.

11. Tou, Julius T. Digital and Sampled Data Control Systems. Mcgraw-Hill Book Company, Inc. New York. 134-135. 1970.

12. Franklin, Gene F. and J. David Powell. Digital Control of Dynamic Systems. Addison-Wesley Publishing Company, Reading, Massachusetts. 85. 1980.

13. Manley, R. G. Waveform Analysis. John Wiley & Sons Inc. New York. 1945.

14. Barfield, Finley, Flight Dynamics Division, Personal Interview. AFTI/F-16 ADPO, Wright-Patterson AFB OH, 6 June through 31 August 1987.

15. D'Azzo, John J. and C. Houpis. Linear Control System Analysis and Design. Mcgraw-Hill book Company. New York. 1981.

VITA

Captain David M. Thomas [redacted] [redacted] He graduated from high school in Zion, Illinois, in 1974 and attended Milwaukee School Of Engineering for one year. On 22 December 1975, he enlisted in the USAF and upon graduation from Basic Military Training was assigned to Minot AFB, North Dakota where he worked as a B-52 crew chief. He was then selected for the Airman's Education and Commissioning Program and attended the University of Illinois, from which he received the degree of Bachelor of Science in Electrical Engineering in May 1981. Upon graduation, he attended Officers Training School and was commissioned in the USAF on 28 August 1981. He was stationed at Edwards AFB, California, where he worked as a flight test instrumentation engineer until entering the School of Engineering, Air Force Institute of Technology, in May 1986.

[redacted]

[redacted]

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/ENG/87D-66	5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION School of Engineering	6b. OFFICE SYMBOL *(If applicable)* AFIT/ENG	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS *(City, State, and ZIP Code)* Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433		7b. ADDRESS *(City, State, and ZIP Code)*
8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
See Box 19

12. PERSONAL AUTHOR(S)
David M. Thomas, B.S., CPT, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(Year, Month, Day)*	15. PAGE COUNT
MS Thesis	FROM ________ TO ______	1987 December	113

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP
01	04	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Subharmonic Aliasing, Digital Flight Control, AFTI/F-16, Digital Control

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Title: SUBHARMONIC ALIASING AND ITS EFFECTS ON THE AFTI/F-16 DIGITAL FLIGHT CONTROL SYSTEM

Thesis Chairman: Zdzislaw H. Lewantowicz, Lieutenant Colonel, USAF

Deputy Head of Department of Electrical and Computer Engineering

Approved for public release: IAW AFR 190-1.
Lynn Wolaver 14 Mar 88
LYNN E. WOLAVER
De[illegible] for [illegible] Development
Air Force [illegible]
Wright-Pat[illegible]

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE *(Include Area Code)*	22c. OFFICE SYMBOL
Z. H. Lewantowicz, Lt Col, USAF	(513) 255-3576	AFIT/ENG

Previous editions are obsolete.

Abstract

The purpose of this research is threefold. First, determine the cause of subharmonic aliasing, described by the AFTI/F-16 engineers as "the creation of uncorrelated low frequencies whenever a subharmonic of the sample frequency is input into the system". Second, model the subharmonic aliasing effect, so that, by knowing only the input frequency and the system sample rate the output characteristics can be calculated. And third, demonstrate by simulation the effect of input and output filters on the subharmonic alias, and the effect of signals in the subharmonic range ($\omega_s/10 < \omega_0 < \omega_s/2$) on the interchannel difference and the software rate limiter.

The model determined that subharmonic aliasing is the result of impostor frequencies (much like aliasing) being introduced into the output signal by the sampling process. Well defined subharmonic aliases occur due to: impostor frequencies and a phenomena known as apparent low frequency surge, which occurs when the input frequency is nearly an integer multiple (>1) of the impostor frequency.

Simulations demonstrate the effectiveness of input analog low pass filters for attenuating signals in this frequency range. Unfortunately, according to simulations, interchannel differences may exceed Input or Output selector monitor threshold levels for frequencies in the subharmonic range. Simulations also show that it is the combination of well defined subharmonic aliases, the high frequencies (relative to the sample rate) of the input signal, and the nonlinear characteristics of the software rate limiter which form the low frequency output from the rate limiter.

In conclusion, since subharmonic aliasing occurs for frequencies which are high relative to the sample rate, careful selection of the sample rate (and computational frame rate) should reduce this undesireable effect. Additionally, since the phenomenon is actually the combination of two high frequencies, the use of analog lowpass filters at the input does attenuate the subharmonic aliasing effect.